AD-A118 518 DELAWARE UNIV NEWARK DEPT OF CIVIL ENGINEERING F/G 8/3
NUMERICAL MODELING OF THE NEARSHORE REGION.(U)
JUN 82 J T KIRBY, R A DALRYMPLE N00014-81-K-0297
UNCLASSIFIED CE-82-24 NL

1 OF 2
AD A
118518

NUMERICAL MODELING OF THE NEARSHORE REGION

by

James T. Kirby, Jr.

and

Robert A. Dalrymple

Technical Report No. 11

Contract No. N00014-81-K-0297

with the OFFICE OF NAVAL RESEARCH, GEOGRAPHY PROGRAMS

Research Report CE-82-24

June 1982

OCEAN ENGINEERING PROGRAM

DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY OF DELAWARE
NEWARK, DELAWARE
19711

NUMERICAL MODELLING OF THE NEARSHORE REGION

by

JAMES T. KIRBY, Jr.

and

ROBERT A. DALRYMPLE

Technical Report No. 11

Contract No. N00014-81-K-0297

with the Office of Naval Research, Geography Programs

RESEARCH REPORT CE-82-24

June 1982

Ocean Engineering Program

Department of Civil Engineering

University of Delaware

Newark, Delaware

19711

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: ONR TR. No. 11	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Numerical Modelling of the Nearshore Region		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): James T. Kirby, Jr. and Robert A. Dalrymple		8. CONTRACT OR GRANT NUMBER(s): N00014-81-K-0297
9. PERFORMING ORGANIZATION NAME AND ADDRESS		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Department of Civil Engineering, University of Delaware, Newark, DE 19711		12. REPORT DATE: June 1982
		13. NUMBER OF PAGES: 163
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This report has been approved for public release and sale; its distribution is unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Numerical models, nearshore circulation, longshore currents, rip currents

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The thoeretical background and numerical formulation for two finite-difference models for predicting nearshore circulation are reviewed. The models are based on the time- and depth-averaged equations of continuity and momentum, with momentum flux terms derived from linear wave theory. Wave shoaling and refraction are determined by the method of Noda et al. (1974). The first model,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

referred to as linear, neglects the effects of convective acceleration and lateral mixing, and calculates bottom friction using the small mean current assumption. The second, nonlinear model includes the neglected effects, and uses the full bottom stress relation as given by Liu and Dalrymple (1978). Both models solve iteratively for mean currents and mean free surface displacement at discrete grid points.

A calibration of both models is described based on comparison with field data obtained from the NSTS Torrey Pines experiment (Gable, 1979). Finally, various results obtained during previous studies utilizing the models are presented and discussed.

PD-AAB-518

Blank Page

P. ii, vi
P. 28, 44, 76, 110, 114, 120, 122

ABSTRACT

The theoretical background and numerical formulation for two finite-difference models for predicting nearshore circulation are reviewed. The models are based on the time- and depth-averaged equations of continuity and momentum, with momentum flux terms derived from linear wave theory. Wave shoaling and refraction are determined by the method of Noda et al. (1974). The first model, referred to as linear, neglects the effects of convective acceleration and lateral mixing, and calculates bottom friction using the small mean current assumption. The second, nonlinear model includes the neglected effects, and uses the full bottom stress relation as given by Liu and Dalrymple (1978). Both models solve iteratively for mean currents and mean free surface displacement at discrete grid points.

A calibration of both models is described based on comparison with field data obtained from the NSTS Torrey Pines experiment (Gable, 1979). Finally, various results obtained during previous studies utilizing the models are presented and discussed.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter I

INTRODUCTION

The need for methods to accurately predict the magnitude and spatial distribution of nearshore currents is central to the present research efforts aimed at quantifying the transport of marine sediment in the beach and nearshore environment. Investigators have made significant strides in describing the mean wave-induced motions in the surf zone in the twenty years since the correct formulation of the averaged equations of motion. Using the concept of wave momentum flux, or radiation stress (see Longuet-Higgins and Stewart, 1964), it has become possible to analytically predict the wave set-up, longshore currents, and spacing of rip currents on beaches of simple planform. Recent strides have also been made towards predicting the dynamic response of the surf zone to fluctuating driving forces with time scales larger than that of the incident wind waves, such as surf beat (see, for example, Symonds, Huntley, and Bowen, 1982).

In general, however, the nearshore environment is a complex system which is not amenable to analytic treatment. Even in simple situations, the presence of a physical feature predicted by one mechanism, such as the longshore current, will greatly complicate the prediction of a separate feature, such as rip currents. In addition, the action of waves and currents on the bottom can create complex topographies, making a reduction of the analytic problems to one space dimension impossible.

This report represents a review and conclusion of several studies conducted at the University of Delaware with the aim of providing a numerical

model for calculating nearshore wave-induced currents and mean water level fluctuations. The purpose of constructing a numerical model rests on the need to extend our predictive capabilities into situations which lie beyond the scope of analytic methods. In the end, all numerical models, as well as analytic formulations, are limited in scope by the simplifying assumptions incorporated in their theoretical framework; in this regard, the present models represent an attempt to extend present analytic treatments to the case of a complex topography in two dimensions. The models do not consider the associated sediment transport problem, although this capability can be added (see McDougal, 1979, and Paddock and Ditmars, 1981). Also, the models require that the incident wave field be regarded as monochromatic, or, after some model modifications, narrow banded enough to be represented as a modulated wave train at a single carrier frequency.

The models described here utilize a wave refraction scheme developed by Noda et al. (1974). Using this scheme, Birkemeier and Dalrymple (1976) developed a circulation model, referred to here as the linear model, which neglected the effect of convective accelerations and lateral mixing. The model of Ebersole and Dalrymple (1979), referred to here as the nonlinear model, extended the treatment to include these effects.

In Chapter II, the theoretical framework for the two models is described, followed by an outline of the numerical formulations in Chapter III. In Chapter IV, we present a calibration of the models using field data from the NSTS Torrey Pines experiment (Gable, 1979). Chapter V gives a summary of applications of the models presented in Birkemeier and Dalrymple (1976) and Ebersole and Dalrymple (1979).

Chapter II

THEORETICAL DEVELOPMENT OF THE MODELS

1. INTRODUCTION

In this chapter the theoretical framework for the development of time averaged governing equations for the problem of waves and currents in the nearshore zone is outlined. The development of the numerical circulation models in either the "linear" or "nonlinear" form is then described in Chapter III based on the theoretical framework.

In general, the development of each model has been described in previous technical reports; the linear model in Birkemeier and Dalrymple (1976), and the nonlinear model in Ebersole and Dalrymple (1979). For this reason, some of the derivations and intermediate steps needed to develop the governing equations are not described in detail in the present report. The reader can refer to the previous work for missing details. However, both models now include the option of calculating wave energy decay due to interaction with the bottom, which has not been included in previous reports. The theory and implementation of this option is described in detail.

2. THEORETICAL FRAMEWORK FOR THE MODELS

The basis for any fluid dynamic model rests on the principle of conservation of mass, conservation of momentum (the Navier-Stokes equations),

and conservation of energy. The resulting system of equations, together with boundary conditions which quantify the interaction of the fluid continuum with its solid and free bounding surfaces, give a mathematical representation of the physical problem of interest. The problem may then be further simplified by assumptions which are consistent with the physical processes involved.

Here, we are interested in the effect of waves propagating towards shore over a complex bathymetry and breaking, and the mean currents driven by changes in the flux of wave momentum. The problem has two apparent time-scales; a fast time scale associated with the oscillation of the incident waves, and a slower time scale over which the characteristics of the incident wave, such as height and angle of incidence, may vary. The longer time scale may also include the effect of changes in wind, tidal oscillation, and, in models, which include sediment transport, gradual shift of the bottom. Since our attention here is towards mean quantities which are reasonably steady in time, the set of equations may be averaged over the faster time scale of the wave oscillation to remove the direct effect of the oscillation. The effect of the waves then is represented by a stress-like term acting on the slowly varying mean flow pattern. In addition, since we are mainly interested in net transport quantities rather than detailed structure of the velocity profiles over depth, the equations may be averaged over depth, reducing the entire problem to a two dimensional problem in the horizontal plane together with appropriate boundary conditions. This averaged model can then be solved using numerical procedures, as discussed in Chapter III. The quantities to be determined are: the horizontal components of mean wave-induced current,

the local wave height and angle of incidence, and the set-up, or wave-induced deviation of the mean water surface from its still-water level.

2.1 Specification of the Boundary Value Problem

A right-handed coordinate system is defined with x in the offshore direction, normal to the coastline, y in the longshore direction, and z vertically upward (see Figure 1). The continuity equation is

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0 \tag{2.1}$$

where ρ is the water density and (u,v,w) are the (x,y,z) components of velocity, respectively. The continuity equation can be further reduced to the form

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.2}$$

consistent with an assumption of constant water density ρ.

The momentum equations are, in the x direction

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{1}{\rho}\left\{\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right\} , \tag{2.3}$$

in the y direction,

$$\frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial vw}{\partial z} = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \frac{1}{\rho}\left\{\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z}\right\} , \tag{2.4}$$

and in the z direction,

$$\frac{\partial w}{\partial t} + \frac{\partial uw}{\partial x} + \frac{\partial vw}{\partial y} + \frac{\partial w^2}{\partial z} = -\frac{1}{\rho}\frac{\partial}{\partial z}(P + gz) + \frac{1}{\rho}\left\{\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z}\right\}, \tag{2.5}$$

after substitution of the continuity equation into the convective acceleration terms.

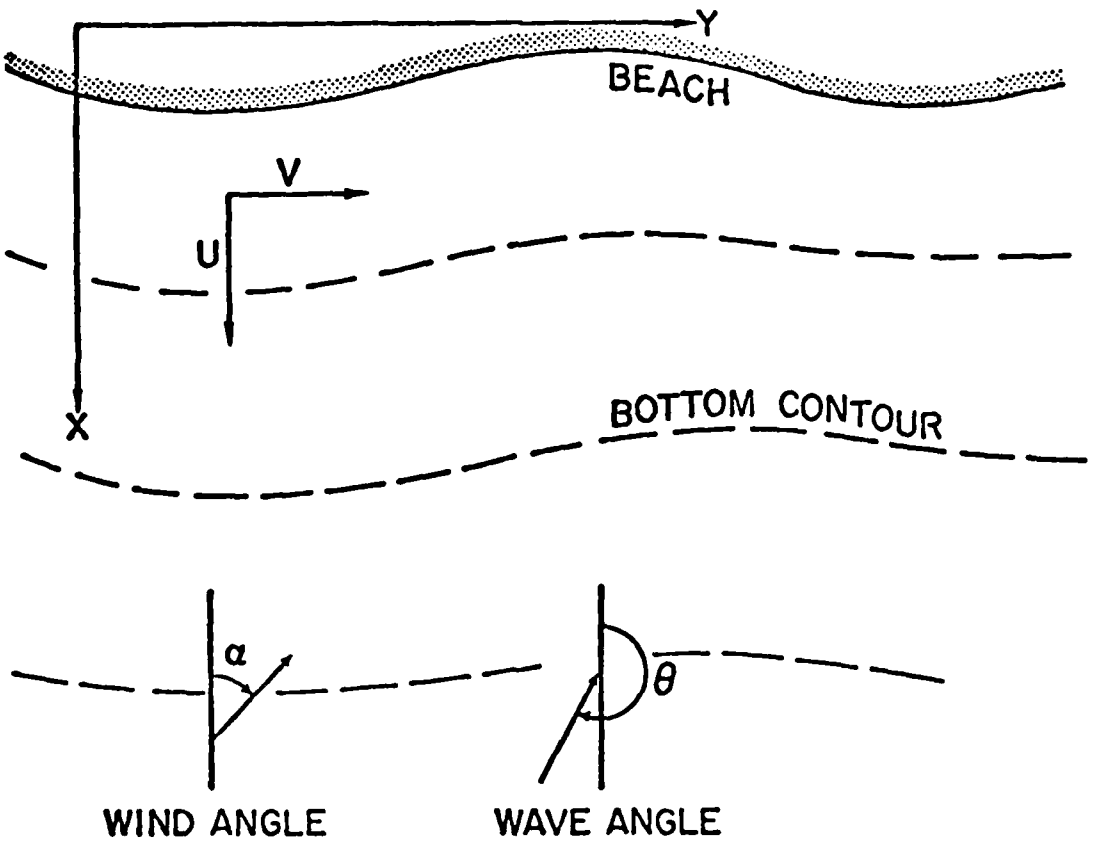


Figure 2-1. Plan Definition Sketch For Coordinate System.

Boundary Conditions

Certain boundary conditions are required at the physical boundary of the water body in question in order to correctly specify the problem. First, kinematic conditions are specified at the free surface and rigid bottom, which state that water particles may not cross the boundary surface, whether it be rigid or moving.

The equation of a surface of the fluid domain is given by

$$F(x,y,z,t) = 0 \quad .$$

A water particle cannot flow across the surface, otherwise the surface would cease to exist. Mathematically, this condition is expressed by the total time rate of change of the function $F(x,y,z,t)$ being equal to zero.

$$\frac{D}{Dt}\,(F(x,y,z,t)) = 0 \quad .$$

At the free surface the boundary is given by

$$F_1(x,y,z,t) = z - \eta(x,y,t) = 0$$

and at the bottom

$$F_2(x,y,z,t) = z + h(x,y,t) = 0 \quad .$$

Therefore, for the kinematic free surface boundary condition (KFSBC),

$$\frac{\partial F_1}{\partial t} + u\,\frac{\partial F_1}{\partial x} + v\,\frac{\partial F_1}{\partial y} + w\,\frac{\partial F_1}{\partial z} = 0 \qquad z = \eta$$

or

$$\frac{\partial \eta}{\partial t} + u_\eta\,\frac{\partial \eta}{\partial x} + v_\eta\,\frac{\partial \eta}{\partial y} - w_\eta = 0 \quad . \tag{2.6}$$

For the bottom boundary condition (BBC) we get,

$$\frac{\partial h}{\partial t} + u_{-h}\frac{\partial h}{\partial x} + v_{-h}\frac{\partial h}{\partial y} + w_{-h} = 0 \qquad (2.7)$$

where u,v,w are the velocity components in the x,y and z directions and the subscripts denote the evaluation of a specific term at the bottom, $z = -h$, or at the free surface, $z = \eta$.

In addition to the kinematic free surface boundary condition, a dynamic free surface boundary condition is required as well, which states that

$$P = \text{constant} \qquad z = \eta$$

This condition is satisfactory if we neglect the local generation of waves by wind or the deformation of the free surface due to barotropic effects. Then, using Bernoulli's equation expanded to the free surface, we obtain

$$-\frac{\partial \phi_\eta}{\partial t} + \frac{(u_\eta^2 + v_\eta^2 + w_\eta^2)}{z} + g\eta = 0 \qquad z = \eta \qquad (2.8)$$

after setting $P = 0$, where $\phi(x,y,z,t)$ is the velocity potential for the wave motion.

The model also requires lateral boundary conditions. In the y (longshore) direction, the bathymetry, given by the surface h(x,y) will be required to be periodic. Since the offshore wave conditions will be assumed uniform, lateral periodicity conditions for wave and currents are also assumed.

Offshore in the x direction, the usual boundary condition for a wave problem would be to assume that all waves at the boundary other than the

incident wave are propagating away from shore, the radiation condition (Sommerfeld (1949)). However, the circulation model does not directly calculate an actual wave field. Rather, a somewhat arbitrary condition,

$$u = 0 \quad ; \quad x = \text{furthest offshore grid}$$

is imposed, which serves to put a bound on the horizontal extent of the flow under consideration.

Specification of an onshore boundary condition is an uncertain task, due to the complexity of the surf zone. In general, it is likely that some wave energy survives the breaking process and reflects from the beach, leading to waves propagated back into the region of the model. However, it is assumed for the purpose of modelling that all wave energy decays in the surf zone, reducing the wave height to a value of zero at the shoreline. In addition, both u and v are set equal to zero at the shoreline.

2.2 Depth and Time Averaged Forms of the Equations

By integrating the equations of motion and continuity over depth and substituting the boundary conditions, the problem is reduced to equations in two horizontal dimensions, with lateral boundary conditions prescribed. Secondly, the quantities of principal interest are average in nature, i.e., mean currents, wave height and wave angle, and mean water level. The equations can then be time averaged over a wave period to remove consideration of instantaneous wave induced motions. The quantities remaining would be free to vary slowly in time in response to changing offshore conditions.

Integrating Eq. (2.2) over depth from $z = -h(x,y,t)$ to $z = \eta(x,y,t)$, using *Leibnitz rule (Hildebrand (1976), p. 365) to remove partial derivatives* from within the integrals, and substituting Eqs. (2.5) and (2.6), the continuity equation becomes

$$\frac{\partial}{\partial t} \int_{-h}^{\eta} \rho dz + \frac{\partial}{\partial x} \int_{-h}^{\eta} \rho u dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} \rho v dz = 0 \qquad (2.9)$$

Let both u and v be comprised of a time independent mean flow and a wave induced flow.

$$u = \overline{U} + \hat{u}$$

$$v = \overline{V} + \hat{v} \quad .$$

By substituting the above expressions for u and v into Eq. (2.7) and time averaging over one wave period so as to eliminate the wave induced fluctuations, the continuity equation can be written as

$$\frac{\partial}{\partial t} \{\rho(h + \overline{\eta})\} + \frac{\partial}{\partial x} \{\rho\overline{U}(h + \overline{\eta})\} + \frac{\partial}{\partial x} \overline{\int_{-h}^{\eta} \rho \hat{u} dz}$$

$$+ \frac{\partial}{\partial y} \{\rho\overline{V}(h + \overline{\eta})\} + \frac{\partial}{\partial y} \overline{\int_{-h}^{\eta} \rho \hat{v} dz} = 0 \qquad (2.10)$$

where the symbol "$\overline{\quad}$" denotes the time average over one wave period and $\overline{\eta}$ the time independent mean free surface displacement. Note that the time averages of the vertically integrated wave induced velocities $\hat{u}$ and $\hat{v}$ are not zero.

Defining $\quad U \equiv \overline{U} + \tilde{u}$

$$V \equiv \overline{V} + \tilde{v} \quad ,$$

where $\tilde{u} = \overline{\dfrac{\int_{-h}^{\eta} \rho \hat{u} dz}{\rho(h + \bar{\eta})}}$ and $\tilde{v} = \overline{\dfrac{\int_{-h}^{\eta} \rho \hat{v} dz}{\rho(h + \bar{\eta})}}$

are the wave induced mass transport velocities, and substituting the total depth D for $(h + \bar{\eta})$, the time averaged, depth integrated continuity equation is, in its final form

$$\frac{\partial \bar{\eta}}{\partial t} + \frac{\partial}{\partial x}(UD) + \frac{\partial}{\partial y}(VD) = 0 \quad . \tag{2.11}$$

Inherent in the derivation are the assumptions that the bottom is constant with time and the density is constant in space and time.

Momentum Equations

The x and y momentum equations are manipulated in the same way as the continuity equation in order to achieve equations which are independent of wave induced oscillations, i.e., they are integrated over depth and time averaged over a wave period. Details of this derivation may be found in Ebersole and Dalrymple (1979). The resulting equations are equivalent to those given by Phillips (1977), but are written explicitly in terms of depth averaged mean velocities. The equations contain terms for mean bottom shear stress, mean surface shear stress due to wind, mean lateral friction (not used in the linear model), and excess mean momentum stress due to wave action (the radiation stress, see Longuet-Higgins and Stewart (1964)). The x momentum equation in its final form can be written as,

$$\frac{\partial}{\partial t}(UD) + \frac{\partial}{\partial x}(U^2 D) + \frac{\partial}{\partial y}(UVD) = -gD\frac{\partial \bar{\eta}}{\partial x} - \frac{D}{\rho}\frac{\partial \bar{\tau}_{\ell}}{\partial y}$$

$$- \frac{1}{\rho}\frac{\partial S_{xx}}{\partial x} - \frac{1}{\rho}\frac{\partial S_{xy}}{\partial y} + \frac{1}{\rho}\overline{\tau}_{sx} - \frac{1}{\rho}\overline{\tau}_{bx} \tag{2.12}$$

and the final form of the y momentum equation can be written as

$$\frac{\partial}{\partial t}(VD) + \frac{\partial}{\partial x}(UVD) + \frac{\partial}{\partial y}(V^2D) = -gD\frac{\partial \bar{\eta}}{\partial y} - \frac{D}{\rho}\frac{\partial \bar{\tau}_\ell}{\partial x}$$

$$- \frac{1}{\rho}\frac{\partial S_{xy}}{\partial x} - \frac{1}{\rho}\frac{\partial S_{yy}}{\partial y} + \frac{1}{\rho}\overline{\tau_{sy}} - \frac{1}{\rho}\overline{\tau_{by}} \quad . \tag{2.13}$$

Wave Energy Equation

Following Phillips (1977), an equation expressing the conservation of averaged wave energy may be written as

$$\frac{\partial E}{\partial t} + \frac{\partial}{\partial x}\{E(U + C_g \cos\theta)\} + \frac{\partial}{\partial y}\{E(V + C_g \sin\theta)\}$$

$$+ S_{xx}\frac{\partial U}{\partial x} + S_{xy}\left\{\frac{\partial V}{\partial x} + \frac{\partial U}{\partial y}\right\} + S_{yy}\frac{\partial V}{\partial y} = \varepsilon \tag{2.14}$$

Here, C_g is the wave group velocity and θ is the local wave angle. The quantity ε represents the dissipation of wave energy, and is identically zero in a conservative wave field. It can be given a non-zero value to include the effect of wave damping, as described below.

2.3 Radiation Stresses

The radiation stresses included in Eqs. (2.12), (2.13) and (2.14) represent the stress on the water column induced by wave action. Neglecting the effects of small scale turbulent velocities, the stresses have been given in simple form by Longuet-Higgins and Stewart (1964) as

$$S_{xx} = E\,[(2n - 1/2)\cos^2\theta + (n - 1/2)\sin^2\theta] \tag{2.15}$$

$$S_{yy} = E\,[(2n - 1/2)\sin^2\theta + (n - 1/2)\cos^2\theta] \tag{2.16}$$

$$S_{xy} = S_{yx} = \frac{E}{2}\, n \sin(2\theta) \tag{2.17}$$

where E is the wave energy, θ is the wave angle, and n = ratio of group velocity (C_g), to wave celerity (C). To second order for a progressive small amplitude wave, E and n are given by

$$E = \frac{1}{8}\rho g H^2 \tag{2.18}$$

$$n = \frac{C_g}{C} = \frac{1}{2}\left[1 + \frac{2kh}{\sinh(2kh)}\right] \tag{2.19}$$

k = wave number $(= \frac{2\pi}{L})$

h = depth

L = wave length

H = wave height.

2.4 Wind Stress

Although other methods exist for computing the surface stress due to the wind (see Wu (1968)), the one suggested in the *Shore Protection Manual* (1977) has been utilized. This form was first developed by Van Dorn (1953) and gives a fairly good fit to the existing experimental data. The form of the surface stress is quadratic in the wind speed and is given by

$$\bar{\tau}_{sx} = \rho K |W| W_x \tag{2.20}$$

$$\bar{\tau}_{sy} = \rho K |W| W_y \tag{2.21}$$

where W is the wind speed, and W_x, W_y are wind velocity components in the x and y directions, as determined by the wind angle, α.

The wind stress coefficient K is determined empirically to be dependent on the magnitude of the wind velocity such that

$$K = \begin{cases} K_1 & W \le W_{cr} \\ K_1 + K_2 \, (1 + W_{cr}/W)^2 & W > W_{cr} \end{cases} \tag{2.22}$$

and $K_1 = 1.1 \times 10^{-6}$; $K_2 = 2.5 \times 10^{-6}$

$$W_{cr} = 7.2 \text{ meters/second}$$

A comparison of the wind stress coefficient given here with experimental data is given in Pearce (1972).

2.5 Bottom Stress

The correct method for specifying the effect of a rigid bottom on waves and currents is still a matter of lively debate. For the purpose of modelling hydrodynamics in the nearshore zone, the average bottom shear stress, τ_b, is generally taken to be of the form

$$\overline{\tau_b} = \rho \frac{f}{8} \overline{u_b |u_b|} \tag{2.23}$$

where f is a Darcy-Weisbach friction factor and u_b is the total instantaneous scalar velocity at the bottom (Longuet-Higgins (1970a)). This relation is known as a quadratic friction law. The components of shear stress in the x and y directions can be given quite generally by (Liu and Dalrymple (1978))

$$\overline{\tau_{bx}} \approx \frac{\rho f}{16\pi} \int_0^{2\pi} (U + u_m \cos\theta \cos\sigma t) \cdot |\vec{u_b}| \, d(\sigma t) \tag{2.24}$$

$$\overline{\tau_{by}} \approx \frac{\rho f}{16\pi} \int_0^{2\pi} (V + u_m \sin\theta \cos\sigma t) \cdot |\vec{u_b}| \, d(\sigma t) \tag{2.25}$$

where u_m is the maximum wave orbital velocity given by

$$u_m = \frac{\pi H}{T \sinh kh} \tag{2.26}$$

and $\overrightarrow{u_b}$ is the vector velocity at the bottom. The nonlinear model of Ebersole and Dalrymple (1979) retains these forms, where the integration is approximated by a 16 term Simpson's rule summation.

The linear model retains the form originally used by Birkemeier and Dalrymple (1976). For this form, the assumption that friction is primarily due to the influence of the wave orbital velocity is used. Making a small mean-current assumption, LeBlond and Tang (1974) show that the bottom stress is anisotropic, with

$$\overline{\tau_{bx}} = \frac{\rho f}{2\pi} u_m U \tag{2.27}$$

$$\overline{\tau_{by}} = \frac{\rho f}{4\pi} u_m V \quad , \tag{2.28}$$

with U, V, u_m, and f as previously defined. A derivation of these equations is given in Birkemeier and Dalrymple (1976), Appendix A.

2.6 Wave Refraction and Shoaling Including Wave-Current Interation

The equations which govern both wave refraction and shoaling as a result of wave-current interaction used in the model are those of Noda et al. (1974). The advantage of Noda's method is that it can predict the wave angles and wave heights at certain points rather than along a wave ray. This procedure lends itself well to use in the finite difference model because calculations are performed at points which lie in the center of rectangular grid elements which are part of a larger grid mesh.

Starting with a progressive linear gravity wave, the free surface can be written as,

$$\zeta(x,y,t) = a(x,y,t)\cos\{\phi(x,y,t)\}$$

where a is the wave amplitude and ϕ is a phase function. A wave number vector can be defined as

$$\vec{k} = \nabla\phi \tag{2.29}$$

and a wave scalar frequency can be defined as

$$\bar{\sigma} = -\frac{\partial\phi}{\partial t} \tag{2.30}$$

Using the mathematical property that the curl of a gradient is identically zero, it is shown that

$$\nabla \times \nabla\phi = 0$$

which implies that

$$\nabla \times \vec{k} = 0$$

This equality states that the wave number vector is irrotational. Assuming $\phi(x,y,t)$ is continuous, then

$$\frac{\partial}{\partial t}(\nabla\phi) = \nabla\frac{\partial\phi}{\partial t} \quad .$$

On substituting Eqs. (2.29) and (2.30) into the above expression, it is found that

$$\frac{\partial\vec{k}}{\partial t} + \nabla\bar{\sigma} = 0 \tag{2.31}$$

which is the classical conservation of waves equation.

For the case of a wave propagating on a current with velocity $\vec{u} = u\vec{i} + v\vec{j}$, it can be shown that the scalar frequency with respect to a stationary reference frame is

$$\tilde{\sigma} = \sigma + k \cdot \vec{U} \quad .$$

The wave frequency with respect to a moving reference frame is given by the dispersion relation,

$$\sigma^2 = gk \tanh kh \quad . \tag{2.32}$$

If it is also assumed that the wave number field is constant in time then, from Eq. (2.31),

$$\nabla(\sigma + \vec{k} \cdot \vec{U}) = 0$$

or

$$\sigma + \vec{k} \cdot \vec{U} = \text{constant} \quad . \tag{2.33}$$

This constant can be evaluated for the case where $\vec{U} = 0$ in which case $\sigma = \frac{2\pi}{T}$ where T is the wave period. Eq. (2.33) then becomes

$$\sigma + \vec{k} \cdot \vec{U} = \frac{2\pi}{T} \quad . \tag{2.34}$$

Using the coordinate system shown in Figure 1 and expanding Eqs. (2.29) and (2.34) into Cartesian coordinates and using the dispersion relation, the equations which govern wave refraction through wave-current interaction are given by

$$\cos\theta \left\{\frac{\partial\theta}{\partial x} - \frac{1}{k}\frac{\partial k}{\partial y}\right\} + \sin\theta \left\{\frac{\partial\theta}{\partial y} + \frac{1}{k}\frac{\partial k}{\partial x}\right\} = 0 \tag{2.35}$$

$$\{gk \tanh(kh)\}^{1/2} + Uk \cos\theta + Vk \sin\theta = \frac{2\pi}{T} \qquad (2.36)$$

where θ, k, h, U and V are all functions that may vary in both the x and y directions.

The shoaling of waves is also affected by the interaction of waves and currents. The effect on the waves is determined by solving the energy equation. Dividing Eq. (2.14) by E and expanding in Cartesian coordinates, we get

$$\frac{1}{E}\frac{\partial E}{\partial t} + (U + C_g \cos\theta)\frac{1}{E}\frac{\partial E}{\partial x} + (V + C_g \sin\theta)\frac{1}{E}\frac{\partial E}{\partial y}$$

$$+ \frac{\partial}{\partial x}(U + C_g \cos\theta) + \frac{\partial}{\partial y}(V + C_g \sin\theta)$$

$$+ \frac{1}{E}\left\{S_{xx}\frac{\partial U}{\partial x} + S_{xy}\frac{\partial U}{\partial y} + S_{yy}\frac{\partial V}{\partial y} + S_{xy}\frac{\partial V}{\partial y}\right\} = \frac{\varepsilon}{E} \quad .$$

Using this result, carrying out the differentiation, and letting a quantity Q be defined as

$$Q = \frac{1}{E}\left\{S_{xx}\frac{\partial U}{\partial x} + S_{xy}\frac{\partial U}{\partial y} + S_{xy}\frac{\partial V}{\partial x} + S_{yy}\frac{\partial V}{\partial x}\right\}$$

the energy equation becomes,

$$\frac{2}{H}\frac{\partial H}{\partial t} + (U + C_g \cos\theta)\frac{2}{H}\frac{\partial H}{\partial x} + (V + C_g \sin\theta)\frac{2}{H}\frac{\partial H}{\partial y} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y}$$

$$- C_g \sin\theta\frac{\partial\theta}{\partial x} + \cos\theta\frac{\partial C_g}{\partial x} + C_g \cos\theta\frac{\partial\theta}{\partial y} + \sin\theta\frac{\partial C_g}{\partial y} + Q = \frac{\varepsilon}{E} \quad . \qquad (2.37)$$

For all applications of the model the wave height H is assumed constant in time, so $\frac{\partial H}{\partial t} = 0$. From linear wave theory the group velocity C_g is given by

$$C_g = \frac{C}{2}\left\{1 + \frac{2kh}{\sinh(2kh)}\right\}$$

where

$$C = \left\{\frac{g}{k}\tanh(kh)\right\}^{1/2}$$

is the wave speed or celerity, k is the wave number, and h is the water depth.

2.7 Wave Breaking Criteria

Since Eq. (2.37) is applicable only in determining the wave heights of nonbreaking waves, some method is needed to determine the point of breaking and the wave heights after breaking. Though a number of formulas for doing this have been developed, there is not as yet one which is universally applicable or accepted. The choice of a breaking criteria, although somewhat arbitrary, must be made with care since it determines the width of the surf zone and thus controls the set-up. The simplest breaking criteria is that predicted by solitary wave theory.

$$\left(\frac{H}{D}\right)_b = \text{constant} = .78 \tag{2.38}$$

where the subscript, b, denotes the value at breaking. There is, however, considerable evidence (Weggel, 1972) that this is an oversimplification.

Noda *et al.* (1974) used a modified version of the Miche formula

$$\left(\frac{H}{L}\right)_b = .12 \tanh\left(\frac{D}{L}\right)_b \tag{2.39}$$

both to predict the point of breaking and the decay of the wave after breaking. This was done by calculating both a wave height from Eq. (2.37) and a breaking height from Eq. (2.39). When the point was reached where the wave height was

equal to or greater than the breaking height, the wave was considered to have broken and the wave height from Eq. (2.39) was used. Eq. (2.39) is used in both models to determine broken wave heights.

2.8 Lateral Mixing

The non-linear model of Ebersole and Dalrymple (1979) includes the effect of the lateral shear stress terms

$$\frac{-D}{\rho}\frac{\partial \overline{\tau_\ell}}{\partial y} \quad , \quad \frac{-D}{\rho}\frac{\partial \overline{\tau_\ell}}{\partial x}$$

in the x and y momentum equations (2.12) and (2.13) respectively. The need for these terms is pointed out by Longuet-Higgins (1970a) in his treatment of the longshore currents due to obliquely incident waves on a plane beach. Neglecting the effects of lateral shear stresses led to prediction of a longshore current distribution with a discontinuity at the breaker line and no current outside the surf zone. However, physical observation in both the laboratory and field indicate that mean longshore flows are present beyond the breaker line. Longuet-Higgins (1970b) presented a formulation which included the effect of lateral mixing as the means for handling the effect of lateral shear; the shear stresses are thus based on turbulent Reynolds stresses proportional to the local gradient of the mean velocity. The resulting velocity distributions have no discontinuity at the breaker line, and the peak velocities are shifted shoreward of the breaker line. Figure 2-2 shows a comparison of a theoretical velocity profile to one without lateral mixing included.

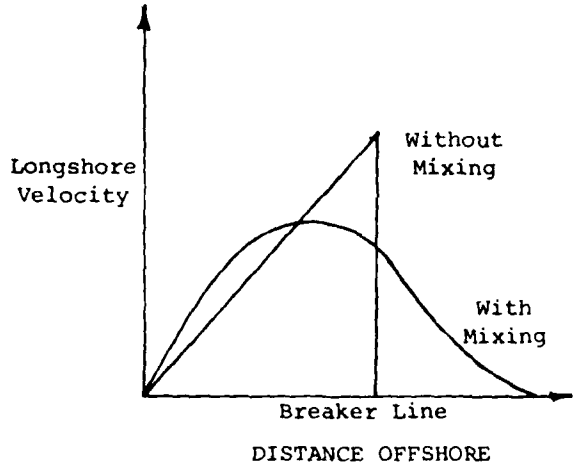


Figure 2-2 Longuet-Higgins' Analytical Solution for Oblique Wave Attack on a Plane Beach

Following the derivation of Longuet-Higgins (1970b), the lateral shear stress is written as

$$\tau_{\ell} = -\rho(\varepsilon_y \frac{\partial U}{\partial y} + \varepsilon_x \frac{\partial V}{\partial x}) \qquad (2.40)$$

Longuet-Higgins argued that the mixing coefficient ε_x must tend to zero as the shoreline was approached since the turbulent eddies responsible for mixing cannot be of a scale greater than the distance to shore. He assumed that ε_x is proportional to the distance offshore, x, multiplied by some velocity scale which he chose to be $\sqrt{gh}$, the speed of a wave in shallow water where h is the local water depth. Therefore, ε_x can be written as

$$\varepsilon_x = Nx\sqrt{gh}$$

where N is a dimensionless constant whose limits Longuet-Higgins gave as

$$0 \leq N \leq 0.016 \quad .$$

In this model the coefficient, ε_x, was allowed to vary linearly with x to some value around the breaker line. From this point seaward the coefficient remained at this constant value. The reason for this approximation is that physically there must be some limit on the scale of these eddies. This limit is at present not known. The coefficient, ε_y, was chosen to be constant. The values of N and ε_y are chosen during the calibration of the model.

2.9 Wave Height Decay

In all applications of the circulation model to date, calculation of wave parameters has been confined to a region within several wavelengths of the shore. Over this distance, wave energy decay due to interaction with the bottom is not likely to be significant, except possibly in the case of waves propagating over a soft mud bottom (Dalrymple and Liu (1978)), which is not treated here. However, the circulation model could reasonably be used to model propagation over much longer offshore extents of shallow coastal waters. At some point, the accumulated effects of bottom interaction would result in wave height reductions of a significant degree. In order to accurately model the amount of wave energy available for driving currents and maintaining mean water level variations, it is necessary to include the effects of various dissipation mechanisms in the equation for calculating wave height (2.37).

Ebersole and Dalrymple (1979) included in the nonlinear computer model, but did not describe, the option to calculate wave energy dissipation due to viscous shear in the bottom boundary layer, and due to the effect of "pumping" of water through the permeable sand bottom due to wave induced pressure

gradients at the water-bottom interface. At present, the option of calculating wave energy decay due to both mechanisms is included in both the linear and nonlinear model.

The application of wave damping to the nearshore circulation model has been described by McDougal (1979), who included the energy dissipation rate ε in Eq. (2.14) in a sediment transport model based on the linear wave-current interaction model. The derivation of ε is based on Liu (1973). Let ε be given by

$$\varepsilon = \varepsilon_p + \varepsilon_f \tag{2.41}$$

where ε_p is the dissipation rate due to the permeability of the bottom, and ε_f is the dissipation rate due to bottom friction. The two effects are treated separately.

Liu (1973) solved the linear problem for waves over a porous bed of infinite depth. If the wave amplitude $a(x,y,t) = \frac{H}{2}(x,y,t)$ is assumed to be a slowly varying function of the form

$$a(x,y,t) = a_o e^{-\alpha t} \quad ,$$

Liu's solution leads to the following form for α;

$$\alpha = \frac{kg}{2\left|\cosh kh + \left(\frac{\sigma}{Q}\right) \sinh kh\right|^2} \left\{ \frac{\nu}{K_p} \frac{1}{|Q|^2} + \sqrt{\frac{\nu}{2\sigma}} \frac{k}{\sigma} \left| \frac{i\sigma}{Q} - 1 \right|^2 \right\} \tag{2.42}$$

where Q is given by

$$Q = \frac{i\sigma}{p_o} - \frac{\nu}{K_p} \tag{2.43}$$

and

K_p = bed permeability $\approx 10^{-10}$ M^2

P_o = bed porosity ≈ 0.6

ν = kinematic viscosity $\approx$

For these values of K_p, P_o, and ν, the viscosity term in Q is dominant, and α may be reduced to the form

$$\alpha = \frac{gk}{2 \cosh^2 kh} \frac{k_p}{\nu} \tag{2.44}$$

The dissipation rate α is related to ε_p by

$$\varepsilon_p = \frac{\partial E}{\partial t} = - 2\alpha E$$

giving

$$\varepsilon_p = - \frac{gkE}{\cosh^2 kh} \frac{K_p}{\nu} \tag{2.45}$$

The rate of energy dissipation due to bottom friction, ε_f, is given by

$$\varepsilon_f = \overline{(\vec{\tau}_b \cdot \vec{U})} A_b \tag{2.46}$$

where A_b is the bed surface area.

Substituting for $\vec{\tau}_b$ and $\vec{U}$ using results derived in the absence of a mean current, we obtain the form

$$\varepsilon_f = - \frac{\rho f}{6\pi} (u_m)^3$$

where u_m is the maximum wave induced velocity at the bottom, given by (2.26).

After some manipulation, we obtain the form

$$\varepsilon_f = -\frac{k}{6\pi}\frac{\sigma f H}{(\cosh^3 kh - \cosh kh)} \cdot E \qquad (2.47)$$

The total energy dissipation including the effects of the porous bottom and friction is given by

$$\varepsilon = \varepsilon_p + \varepsilon_f$$

$$= -\left\{\frac{g k}{\cosh^2 kh}\frac{K_p}{\nu} + \frac{\sigma k f H}{6\pi(\cosh^3 kh - \cosh kh)}\right\} \cdot E \qquad (2.48)$$

The option of including these effects is available in both forms of the numerical model presented here.

3. AN OVERVIEW OF THE LINEAR AND NONLINEAR MODELS

While both of the numerical models described in Chapter III are based on the theory outlined in the previous section, each model contains a somewhat different subset of the overall development. In this section, we review the differences and similarities between the models before going on to their numerical formulation.

The intent of both of the models is to solve for the mean values U, V, and $\bar{\eta}$ at each grid point by solving the time and depth averaged equations of continuity (2.11) and momentum (2.12-13). Each model utilizes the refraction scheme of Noda *et al.* (1974), as represented by (2.35-37), to solve for wave angle and wave height. The model of Birkemeier and Dalrymple (1976) treats linearized forms of Eqs. (2.12-13), obtained by dropping the convective acceleration terms

$$\frac{\partial}{\partial x}(U^2D) \quad ; \quad \frac{\partial}{\partial y}(UVD)$$

from the x-momentum equation, and the terms

$$\frac{\partial}{\partial x}(UVD) \quad ; \quad \frac{\partial}{\partial y}(V^2D)$$

from the y-momentum equation. The model of Ebersole and Dalrymple (1979) retains the full nonlinear form of the momentum equations, leading to the principle theoretical difference between the models as well as explaining the distinction expressed by their names. The mathematical differences in the governing equations also lead to the requirement of significantly different numerical schemes, discussed in Chapter III.

The momentum equations contain various forcing terms which are formulated as stresses or stress gradients; these include radiation, surface and bottom stresses and a lateral stress representing the effect of turbulent mixing. The models treat radiation stresses and surface wind stresses identically. The linear model treats bottom stress according to a "weak current" formulation developed by LeBlond and Tang (1974), based on the assumption that the stress develops principally in response to the wave orbital motion. The nonlinear model uses a more exact bottom stress formulation, making no assumption as to the relative magnitude of wave orbital and mean current velocities. This distinction between the models is more historical than essential; the linear model can be updated to include the more exact relation given by Eqs. (2.24) and (2.25). This modification has been used by Allender et al. (1981) in a version of the linear model described here. A further possibility would be to represent the bottom friction using a "large-current"

formulation developed by Liu and Dalrymple (1978); this extension has not been investigated.

The remaining distinction between the models stems from the neglect of the "lateral mixing" terms Eqs. (2.12-13)

$$-D\frac{\partial \bar{\zeta}}{\partial y} \quad ; \quad \frac{-D}{c}\frac{\partial \bar{\zeta}}{\partial x}$$

in the linear model, and their inclusion in the nonlinear model. This difference is again historical rather than essential; these terms could be included in a modified version of the linear model, although their inclusion would require more effort than the bottom friction modification discussed above. No investigation of the effect of including lateral mixing in the linear model has been made to date.

Chapter III

FORMULATION OF THE NUMERICAL MODELS

1. INTRODUCTION

In the previous chapter a mathematical formulation of the governing equations and associated boundary conditions for the problem of nearshore wave-induced circulation has been outlined. In this chapter, the numerical formulations of the mathematical model are reviewed, corresponding to the work of Birkemeier and Dalrymple (1976) and Ebersole and Dalrymple (1979).

The two models reviewed shall be referred to as the "linear model" and the "nonlinear model." The models contain significant differences as well as significant similarities in their numerical formulation as well as in their underlying mathematical formulation. Differences in the models will be discussed below following a review of the overall structure common to both models.

Both models reviewed here are finite-difference approximations to a set of three first order hyperbolic differential equations consisting of the continuity equation (2.11) and the x and y direction momentum equations (2.12, 2.13), with associated unknowns U(x,y,t), V(x,y,t) and $\bar{\eta}$(x,y,t). The quantities of wave height H(x,y,t) and wave angle θ(x,y,t) are solved for using the refraction scheme Eq. (2.35) and the wave energy equation (2.37). The models as developed contain the flexibility to handle unsteady response to

time varying incident wave conditions; however, we will be concerned exclusively with representing the response to a constant wave climate. In this guise, the models represent iterative schemes to determine the response to a steady state exciting force, and updated unknowns may be regarded as iterated values rather than advanced-in-time values.

Listings of the computer programs for the linear and nonlinear models are given in Appendix I together with some notes on running the programs.

2. THE GRID SCHEME AND LOCATION OF THE UNKNOWN QUANTITIES

The first step in constructing a finite difference model lies in choosing a network of discrete grids overlaying the physical domain of interest. The grid used by the present models is that of Noda et al. (1974), as illustrated in Figure 3-1. The physical domain is divided into regular grids of longshore extent Δy and offshore extent Δx. The topography is assumed to be periodic in the longshore y direction with a length λ given by

$$\lambda = (N - 1) \cdot \Delta y$$

Various requirements affecting the choice of grid are:

1. The grid must extend offshore far enough to remove the offshore region of the domain from the influence of currents driven by the surf zone, and to allow for the specification of a uniform longshore depth which will not significantly alter the wave refraction results in the nearshore.

2. The grid mesh must be fine enough to resolve the surf zone in a reasonable manner.

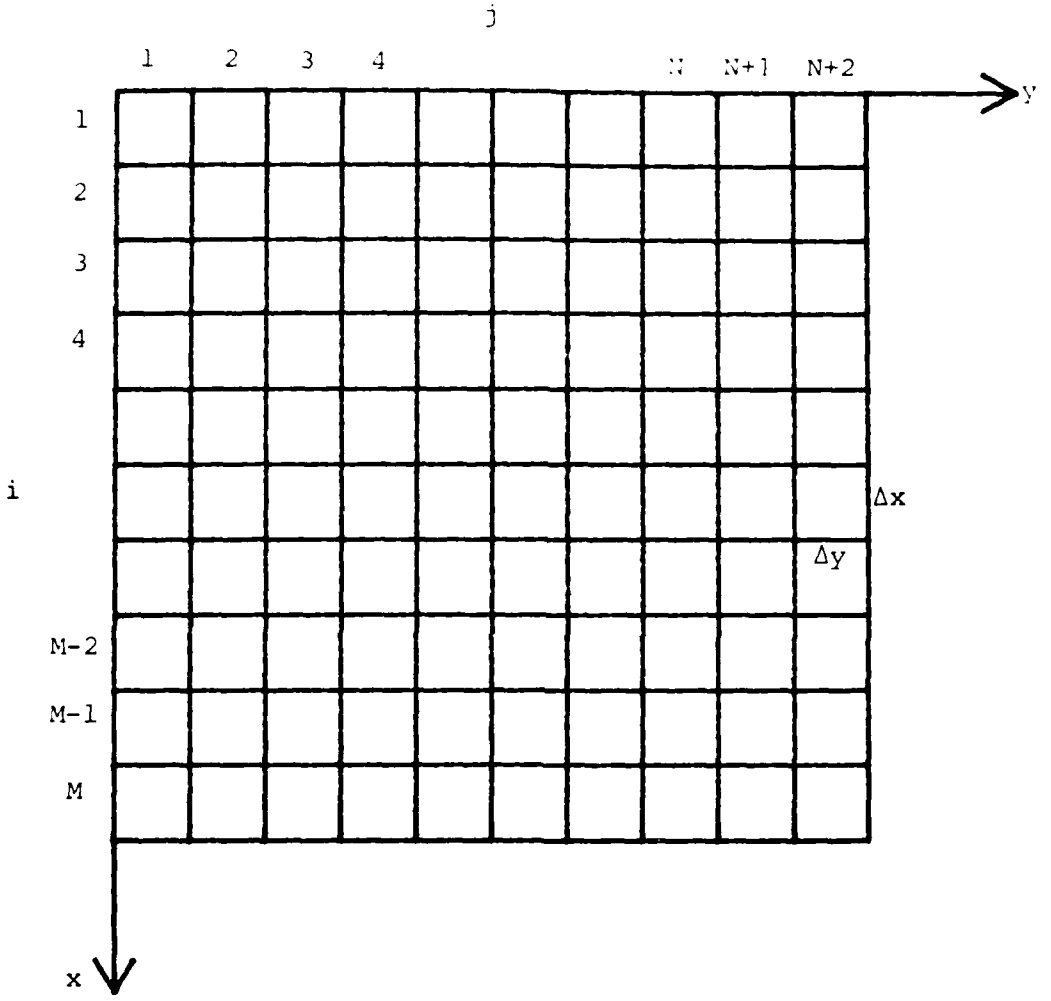


Figure 3-1. Grid Scheme [After Noda et. al. (1974)]

3. In the event that a single physical feature isolated in the longshore direction is to be modelled, the longshore extent of the grid must be sufficiently large to isolate the physical system from the effect of images created by the longshore periodicity requirement.

In the event that large, non-periodic physical features creating significant current patterns seaward of the breaker line are present, requirements (1) and (3) lead to the choice of a large grid, whereas requirement (2) can lead to choice of quite small offshore grid spacings on steep beaches. The resulting grid will often contain a large number of elements, leading to the requirement of large amounts of computer time to solve the complex set of governing equations.

The fixed and variable quantities are defined in relation to the grid structure as shown in Figure 3-2. The quantity $A_{i,j}$ is defined at the grid center and represents any of the set of quantities

$$A_{i,j} = \{H, \theta, k, \bar{\eta}, S, D, \tau_b, \tau_s\}_{i,j} \tag{3.1}$$

Velocities U and V are given at grid sides. This formulation relates naturally to the conservation law scheme of relating changes of a quantity in a bounded region to the fluxes of that quantity across the bounding surface, and is therefore physically correct in an heuristic sense, as well as possessing whatever degree of numerical accuracy consistent with the chosen difference schemes.

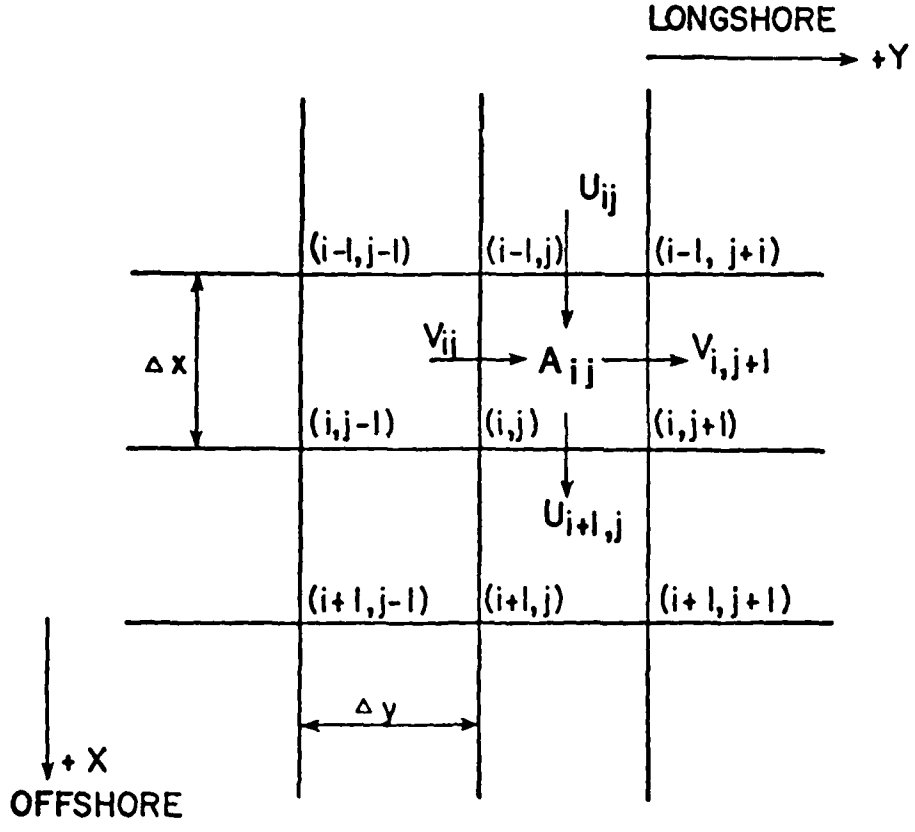


Figure 3-2. Velocity Components for Grid Block (i,j). All Other Variables (D, S_{xx}, S_{xy}, H, etc.) Are Determined at Grid Center.

3. BOUNDARY CONDITIONS

In order to formulate the numerical models, a scheme was established which incorporated the lateral periodicity requirements mentioned above and the no-flow requirements at the shore and the offshore grid row, as mentioned in Chapter II.

The user defined depth grid is established for the M by N portion of the total grid shown in Figure 3.1, with the requirement that

$$D_{i,N} = D_{i,1} \tag{3.2}$$

to satisfy the periodicity requirement. The grid N then extended to the N + 1 and the N + 2 columns according to

$$\begin{aligned} D_{i,N+1} &= D_{i,2} \\ D_{i,N+2} &= D_{i,3} \end{aligned} \tag{3.3}$$

Similar periodicity requirements are met by all calculated quantities (the A_{ij} in Eq. (3.1)). The models calculate values of the A_{ij} quantities from the j = 2 row to the j = N row. Periodicity is then established by setting

$$\begin{aligned} A_{i,1} &= A_{i,N} \\ A_{i,N+1} &= A_{i,2} \\ A_{i,N+2} &= A_{i,3} \end{aligned} \tag{3.4}$$

The velocities U and V are treated somewhat differently in that the calculations

are performed from the j = 3 to the j = N+1 grid rows, with periodicity established accordingly.

At the offshore grid row (i = M), a no-flow condition is applied for U and V. This condition serves to bound the calculated velocities during initial start-up of the model; however, the condition is artificial in that the offshore side of the modelled area becomes a solid vertical boundary. This treatment leads to potential seiching in the model. The presence of a seiching mode in the model calculations has been discussed thoroughly by Birkemeier and Dalrymple (1971) and Ebersole and Dalrymple (1979), who have shown that the period of seiching can be accurately predicted by the shallow water formulas for known topographies. For example, for a basin of triangular cross-section, Wilson (1966) predicts the period of the first mode oscillation by

$$T = \frac{3.28L}{\sqrt{gh_{max}}} \tag{3.5}$$

where

T = period of oscillation in the basin

L = length of the basin = $(M - 1) \cdot \Delta x$

h_{max} = maximum still water depth in the basin.

Similarly, a no flow condition for the shore-normal velocity U is applied at the onshore grid row, whose position remains fixed during a model run. The linear model originally described by Birkemeier and Dalrymple (1976) included a provision for flooding shoreward dry grids in order to model the effect of wave set-up; model results have been seen to be somewhat insensitive to the correction provided by this provision. Flooding is not included in the nonlinear model of Ebersole and Dalrymple or in the present version of the linear model.

In addition to the lateral boundary conditions, the system of hyperbolic equations requires initial conditions for a complete solution. The models are started from a state of rest with no wave field present. In order to reduce the effect of seiching, the wave height H at the offshore grid is brought up to its full value gradually according to

$$H = H_o \tanh \left(\frac{2t}{T}\right) \quad , \tag{3.6}$$

where

t = model time

T = arbitrary fixed time period.

Wave height is typically allowed to build up for 400 model iterations. It is also noted that, in principal, the seiching effect could be removed by setting $\bar{\eta}$ to zero at the offshore boundary, rather than U.

4. FINITE DIFFERENCE OPERATORS AND NOTATIONS

The derivations of the numerical models given by Birkemeier and Dalrymple (1976) and Ebersole and Dalrymple (1979) differ significantly in notation. For this reason, a standardized notation is presented here. We retain as far as possible the terminology of Ebersole and Dalrymple (1979) in describing both the linear and nonlinear models.

The numerical formulations require both averaging and differencing operations. Let F(x,y,t) be an arbitrary function varying in space and time. The average of the function over one grid spacing is given by

$$\overline{F(x,y,t)}^{\,x} \equiv \frac{1}{2}\left\{F\left(x + \frac{\Delta x}{2}, y, t\right) + F\left(x - \frac{\Delta x}{2}, y, t\right)\right\} \tag{3.7}$$

for an average in the x-direction. Successive averaging is given by

$$\overline{\overline{F(x,y,t)}^{\,xy}} \equiv \overline{\overline{F(x,y,t)}^{\,x}}^{\,y} \tag{3.8}$$

First derivatives can be given in terms of forward and backward differences over a single grid space, or as central differences over two grid spaces. We define the forward and backward difference operators, respectively, as

$$\delta_x\{F(x,y,t)\} = \frac{1}{\Delta x}\{F(x + \Delta x,y,t) - F(x,y,t)\} \tag{3.9}$$

$$\delta_{\bar{x}}\{F(x,y,t)\} = \frac{1}{\Delta x}\{F(x,y,t) - F(x - \Delta x,y,t)\} \tag{3.10}$$

and the central difference operator as

$$\delta_{\hat{x}}\{F(x,y,t)\} = \frac{1}{2\Delta x}\{F(x + \Delta x,y,t) - F(x - \Delta x,y,t)\} \tag{3.11}$$

We also define an auxilary operator which represents the central difference at a grid center based on values at the grid sides

$$\delta_{\tilde{x}}\{F(x,y,t)\} = \frac{1}{\Delta x}\left[F\left(x + \frac{\Delta x}{2},y,t\right) - F\left(x - \frac{\Delta x}{2},y,t\right)\right] \tag{3.12}$$

It is easily shown by direct substitution that the $\delta_{\tilde{x}}$ operator is related to $\delta_{\hat{x}}$ through the relation

$$\delta_{\tilde{x}}\{\overline{F(x,y,t)}^{\,x}\} = \delta_{\hat{x}}\{F(x,y,t)\} \tag{3.13}$$

The nonlinear model makes extensive use of the $\delta_{\tilde{x}}$ operator, while the linear model is formulated more conventionally in terms of the standard forward and backward differences given by Eqs. (3.9) and (3.10).

5. FINITE DIFFERENCE FORMS OF THE GOVERNING EQUATIONS - LINEAR MODEL

Before applying the finite difference scheme to the linear formulation, we rewrite the equations here for clarity. The equations of continuity and

x- and y-momentum are given respectively by

$$\frac{\partial \bar{\eta}}{\partial t} = - \frac{\partial}{\partial x} (UD) - \frac{\partial}{\partial y} (VD) \tag{3.14}$$

$$\frac{\partial U}{\partial t} = - g \frac{\partial \bar{\eta}}{\partial x} - \frac{1}{\rho D} \left\{ \frac{\partial S_{xx}}{\partial x} + \frac{\partial S_{xy}}{\partial y} - \overline{\tau_{sx}} + \overline{\tau_{bx}} \right\} \tag{3.15}$$

$$\frac{\partial V}{\partial t} = - g \frac{\partial \bar{\eta}}{\partial y} - \frac{1}{\rho D} \left\{ \frac{\partial S_{xy}}{\partial x} + \frac{\partial S_{yy}}{\partial y} - \overline{\tau_{sy}} + \overline{\tau_{by}} \right\} \tag{3.16}$$

In deriving this set of equations, we have tacitly assumed that variations of $\bar{\eta}$ with respect to time are small in comparison to variations in U and V. As we are striving for a steady state solution, the model results should not be sensitive to this assumption. Care should be taken, however, in removing time derivatives of D in cases where time dependent results are desired.

In keeping with standard techniques for treating first order linear hyperbolic equations, difference equations are formulated using a forward time-forward space (FTFS) scheme for the continuity equation (3.14), and forward time-backward space (FTBS) formulation for the momentum equations (3.15) and (3.16). Recalling that depths and set-up are known at grid centers, while velocities are given at grid boundaries, equations (3.14) - (3.16) are written in finite difference form as:

$$\delta_t\{\bar{\eta}\} = - \delta_x\{U\bar{D}^{\,x}\} - \delta_y\{V\bar{D}^{\,y}\} \tag{3.17}$$

$$\delta_t\{U\} = - g\, \delta_{\bar{x}}\{\eta\} - \frac{1}{\rho \bar{D}^{\,x}} \left\{ \delta_{\bar{x}}\{S_{xx}\} + \delta_{\hat{y}}\{\overline{S_{xy}}\}^{\,x} - \overline{\tau_{sx}}^{\,x} + \overline{\tau_{bx}}^{\,x} \right\} \tag{3.18}$$

$$\delta_t\{V\} = -g\delta_y\{\bar{\eta}\} - \frac{1}{\rho\bar{D}}{}^{y}\left\{\delta_{\hat{x}}\{\overline{S_{xy}}\}^{y} + \delta_y\{S_{yy}\}\right.$$

$$\left. - \overline{\tau_{sy}}^{y} + \overline{\tau_{by}}^{y}\right\} \qquad (3.19)$$

Note that $\delta_t(F)$ yields the forward difference

$$\delta_t(F) = \frac{F^{k+1} - F^k}{\Delta t} \qquad (3.20)$$

where k and k+1 are successive time levels. Expanded forms of the equations (3.17) - (3.19) can be found in Birkemeier and Dalrymple (1976). At each iteration, the values of U and V are updated to time level k+1 using $\bar{\eta}$ and forcing terms evaluated at time level k. Then $\bar{\eta}$ is updated using the newly evaluated values of U and V at time level k+1. The method thus requires only one time level of storage for each of the unknown quantities.

Difference forms of the type used here have been discussed extensively by a number of authors (see, for example, Roache (1976)). The explicit method is subject to certain stability conditions. The problem of obtaining stability criteria for equations with non-constant coefficients is in general unsolved to date; however, we can make a reasonable guess based on constant-coefficient forms of the governing equations. If we drop the forcing terms, assume $D \simeq h$, and hold h constant, we obtain

$$\frac{\partial\bar{\eta}}{\partial t} = -h\left\{\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y}\right\}$$

$$\frac{\partial U}{\partial t} = -g\frac{\partial\bar{\eta}}{\partial x} \qquad (3.21)$$

$$\frac{\partial V}{\partial t} = -g\frac{\partial\bar{\eta}}{\partial y}$$

Cross-differentiating to eliminate U and V, we obtain the second-order hyperbolic equation for $\bar{\eta}$:

$$\frac{\partial^2 \bar{\eta}}{\partial t^2} = gh \left\{ \frac{\partial^2 \bar{\eta}}{\partial x^2} + \frac{\partial^2 \bar{\eta}}{\partial y^2} \right\} \tag{3.22}$$

The stability criterion corresponding to this reduced equation is given by the Courant condition:

$$\Delta t \leq \frac{\Delta x}{\sqrt{2gh}} \quad , \tag{3.23}$$

assuming that $\Delta y = \Delta x$. Generalizing to the full model, the stability criterion (3.23) can be extended to include the effect of rectangular grids and the advection of disturbances by the mean currents.

$$\Delta t \leq \frac{\sqrt{(\Delta x)^2 + (\Delta y)^2}}{\sqrt{2gh} + |\vec{U}|} \tag{3.24}$$

The stability criterion basically states that time steps in the model may not be so large as to allow long wave disturbances to pass completely through a grid in one iteration. Violation of the criterion leads to rapidly growing instability of the numerical solution. In practice, time steps on the order of 0.25 times the maximum Δt are used.

6. FINITE DIFFERENCE FORMS OF THE GOVERNING EQUATIONS - NONLINEAR MODEL

The set of nonlinear momentum equations (2.12) - (2.13), together with the continuity equation (2.11), require a more careful development in difference form than the previously described linear model. Nonlinear formulations are in a practical sense subject to a number of instability mechanisms which are

not strictly related to stability criteria for the corresponding linear formulations, as discussed by Roache (1976).

The nonlinear model has been formulated using a method which has been applied successfully to geophysical and tidal models by Lilly (1965) and Blumberg (1977). Using the difference and average operators defined in section 4, the equations(2.11-13) are rewritten as

Continuity:

$$\delta_t\{\overline{\eta}^{\,t}\} + \delta_x\{\overline{D}^{\,x}U\} + \delta_y\{\overline{D}^{\,y}V\} = 0 \tag{3.25}$$

X-Momentum

$$\delta_t\{\overline{\overline{D}^{\,x}U}^{\,t}\} + \delta_x\{\overline{\overline{D}^{\,x}U}^{\,x}\,\overline{U}^{\,x}\} + \delta_y\{\overline{\overline{D}^{\,y}V}^{\,x}\,\overline{U}^{\,y}\} =$$

$$= -g\,\overline{D}^{\,x}\,\delta_x\{\overline{\eta}\} + \frac{1}{\rho}\{\overline{\overline{\tau_{sx}}^{\,x}} - \overline{\overline{\tau_{bx}}^{\,x}} - \delta_y\{\overline{\overline{S_{xy}}^{\,xy}}\} -$$

$$- \delta_x\{S_{xx}\}\} + \overline{D}^{\,x}\,\delta_y\{\varepsilon_y\delta_y\{U\}\} + \overline{\overline{\varepsilon_x}^{\,xy}}\,\delta_x\{V\}\} \tag{3.26}$$

Y-Momentum

$$\delta_t\{\overline{\overline{D}^{\,y}V}^{\,t}\} + \delta_x\{\overline{\overline{D}^{\,x}U}^{\,y}\,\overline{V}^{\,y}\} + \delta_y\{\overline{\overline{D}^{\,y}V}^{\,y}\,\overline{V}^{\,y}\} =$$

$$-g\,\overline{D}^{\,y}\,\delta_y\{\overline{\eta}\} + \frac{1}{\rho}\left\{\overline{\overline{\tau_{sy}}^{\,y}} - \overline{\overline{\tau_{by}}^{\,y}} - \delta_y\{S_{yy}\}\right.$$

$$\left. - \delta_x\{\overline{\overline{S_{xy}}^{\,xy}}\}\right\} + \overline{D}^{\,y}\,\delta_x\{\varepsilon_y\delta_y\{U\}\} + \overline{\overline{\varepsilon_x}^{\,xy}}\,\delta_x\{V\}\} \tag{3.27}$$

The formula for lateral mixing given by Eq. (2.46) has been substituted into Eqs. (3.26-27). Note that, following Eq. (3.13), the central difference on the time averaged wate set-up in Eq. (3.25) leads to

$$\delta_t\{\bar{\eta}^{t}\} = \delta_t\{\bar{\eta}\} = \frac{\bar{\eta}^{k+1}_{i,j} - \bar{\eta}^{k-1}_{i,j}}{2\Delta t} \tag{3.28}$$

The governing equations in this case then contain function values at three times levels, given by k+1, k, and k-1. The equations (3.25-27) can be found written out with respect to the i,j grid in Ebersole and Dalrymple (1979). These three equations can also be written in the following abbreviated form,

$$\bar{\eta}^{k+1}_{i,j} = \bar{\eta}^{k-1}_{i,j} + 2\Delta t\ F_1 \tag{3.29}$$

$$U^{k+1}_{i,j} = A\ U^{k-1}_{i,j} + 2\Delta t\ F_2 \tag{3.30}$$

$$V^{k+1}_{i,j} = B\ V^{k-1}_{i,j} + 2\Delta t\ F_3 \tag{3.31}$$

where A and B are functions of the depth alone and F_1, F_2, and F_3 are functions of all the variables in the problem. The quantities F_1, F_2, and F_3 contain quantities evaluated at time level k and k-1.

The method of solution for Eqs. (3.29-31) is based on a leapfrog scheme, and thus requires the storage of three time levels of values for all the calculated variables. In order to select the model, a single step is taken based on a forward difference formulation similar to that described in conjunction with the linear model. The forward difference formulation can be indicated schematically by

$$\bar{\eta}^{k+1}_{i,j} = \bar{\eta}^{k}_{i,j} + \Delta t\ F_1 \tag{3.32}$$

$$U^{k+1}_{i,j} = C\ U^{k}_{i,j} + \Delta t\ F_4 \tag{3.33}$$

$$V^{k+1}_{i,j} = D\ V^{k}_{i,j} + \Delta t\ F_5 \quad . \tag{3.34}$$

The computational method is schematized in Figure 3-3.

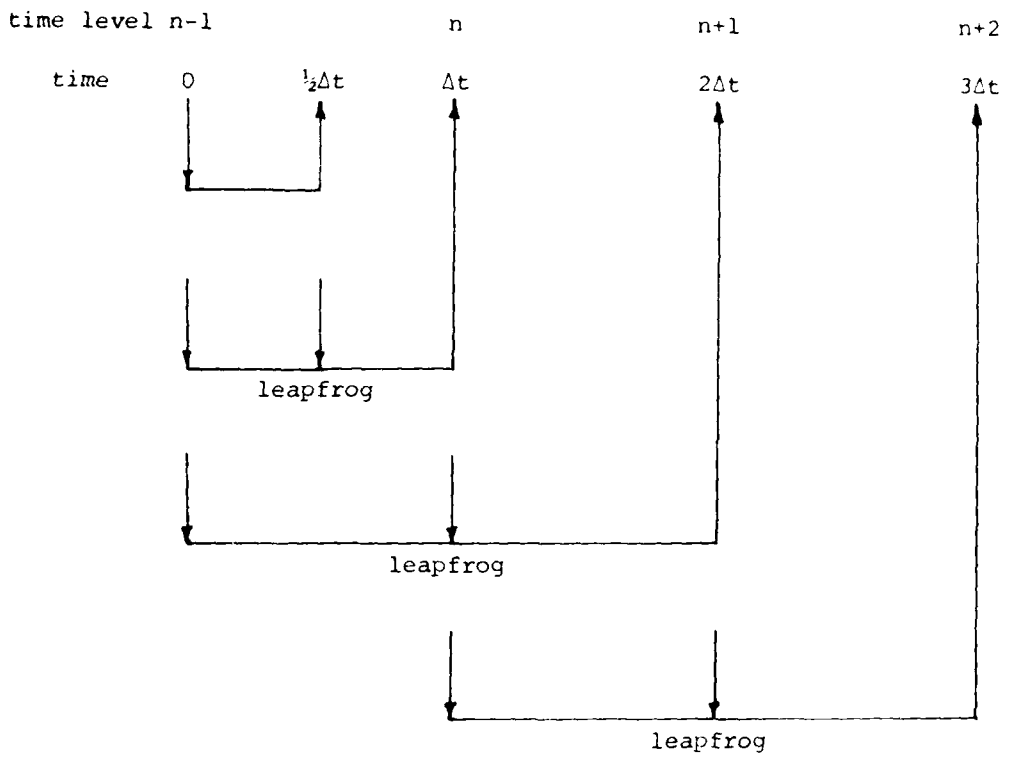


Figure 3-3. General Leapfrog Solution Scheme

The stability criterion for the present scheme was assumed to be given by Eq. (3.24), which has physical if not mathematical significance in the present situation. In practice, as is the case of the linear model, it was necessary to reduce the maximum allowable time step to a value significantly below the one predicted by the stability criterion.

The nonlinear model was also subject to a second computational stability problem. In general, computational schemes for first order equations which are

centered in space and time (CSCT) exhibit forms of unstable behavior. In particular, in the context of the first order parabolic diffusion equation, such explicit CSCT schemes can be shown to be unconditionally unstable. In the present case, as the model approached a steady state, the solution diverged into two disjoint solutions; one associated with the even time steps and the other the odd steps. These solutions oscillated with growing amplitudes about the steady state solution. In Roache (1976), the author referred to this as time splitting.

To alleviate the problem, a leapfrog-backward correction scheme, Kurihara (1965), was initiated every tenth time step. The scheme is shown below as,

$$h^* = h^{k-1} + 2\Delta t\ G^k \tag{3.35}$$

$$h^{k+1} = h^k + \Delta t\ G^* \tag{3.36}$$

where h may be U, V or $\bar{\eta}$. Equation (3.35) is simply the leapfrog calculation done by the Eqs. (3.29-31) where * denotes the new or "interim" time level. Using the new values U, V, $\bar{\eta}$ at time *, the functions G*, like the functions F_1, F_2, and F_3 from Eqs. (3.29-31) are calculated and used in Eq. (3.36) which is merely a backwards difference in time to the level k.

This scheme was chosen because it damps the computational mode of the solution while leaving the physical mode relatively unaffected. For a more in-depth discussion the reader is referred to the work by Kurihara. With this correction scheme included, which essentially "ties" the solutions together every tenth iteration, the model proceeded to reach a steady state without any further time-splitting instability.

7. THE NUMERICAL SCHEME FOR REFRACTION AND THE WAVE HEIGHT FIELD

The equations governing wave refraction, wave height, and wave number are given by Eqs. (2.35), (2,37), and (2.36) respectively. The method of solving these equations for the unknown $\theta_{i,j}$, $H_{i,j}$ and $k_{i,j}$ is identical in both the linear and nonlinear models. The method was given originally by Noda et al. (1974).

If Eq. (2.36) is differentiated with respect to x to get $\frac{\partial k}{\partial x}$ and with respect to y to get $\frac{\partial k}{\partial y}$, these quantities can be substituted into Eq. (2.35), which can then be written in the following form:

$$A \frac{\partial \theta}{\partial x} = B \frac{\partial \theta}{\partial y} = C \tag{3.37}$$

where A, B and C are functions of the quantities sin θ, cos θ, k, h, u and v. By taking a forward difference in x to approximate $\frac{\partial \theta}{\partial x}$ and a backwards difference in y to approximate $\frac{\partial \theta}{\partial y}$, Eq. (3.37) becomes:

$$\theta_{i,j} = D + E\, \theta_{i,j-1} - F\, \theta_{i+1,j} \tag{3.38}$$

where D, E and F are now functions of the quantities sin $\theta_{i,j}$, cos $\theta_{i,j}$, $k_{i,j}$, $h_{i,j}$, $u_{i,j}$, $v_{i,j}$. To evaluate sin $\theta_{i,j}$ and cos $\theta_{i,j}$ Noda used a first order Taylor series expansion to the four neighboring grids (i+1,j), (i-1,j), (i,j+1) and (i,j-1), summed the results and took an average value.

The theta field $\theta_{i,j}$ is solved for in the following way. Snell's Law is used to approximate the angles at the offshore row. Working shoreward Eq. (3.38) is solved for in a row-by-row relaxation until the angles converge to their correct values with wave-current interaction included. After each updated value of theta, a new wave number must be solved for.

Eq. (2.36) can be written as

$$E(k) = \{gk \tanh(kh)\}^{1/2} + u k\cos\theta + v k\sin\theta - \frac{2\pi}{T} = 0 \quad . \tag{3.39}$$

To solve for the wave number, k, after each updated angle is found, the Newton-Raphson Method, or "method of tangents", is used. This method states that

$$k_{new} = k_{old} - \frac{E(k_{old})}{E'(k_{old})}$$

Differentiating Eq. (3.39), k_{new} is iteratively solved for until $|k_{new} - k_{old}| \leq .001|k_{new}|$.

The wave height field is calculated in much the same way as the wave angle field. Multiplying Eq. (2.37) by $\frac{H}{2}$ and letting $\frac{\partial H}{\partial t} = 0$, the energy equation can now be written in the form

$$A \frac{\partial H}{\partial x} + B \frac{\partial H}{\partial y} = C H \tag{3.40}$$

where A, B and C are functions of u, v cos θ, sin θ, C_g, Δx, Δy and the radiation stresses. Taking a forward difference in x to approximate $\frac{\partial H}{\partial x}$ and a backward difference in y to approximate $\frac{\partial H}{\partial y}$ and solving for $H_{i,j}$, it can be shown that

$$H_{i,j} = D\, H_{i,j-1} - E\, H_{i+1,j}$$

where D and E are functions of the same quantities as A, B and C. Again the offshore row of wave heights are obtained from shoaling and refraction due to Snell's Law and the wave height field is determined by relaxing row by row in the shoreward direction. On each row a solution for the wave height is reached when $|H_{new} - H_{old}| \leq .01\, H_{new}$. After each updated value of $H_{i,j}$, a breaking wave height is also calculated from the breaking criteria given by the Miche

formula

$$\left(\frac{H}{L}\right)_b = .12 \tanh(kh)_b \quad .$$

If the calculated $H_{i,j}$ is larger than the allowable breaking height, the height $H_{i,j}$ was set equal to the breaking height.

CHAPTER IV

CALIBRATION OF THE NEARSHORE CIRCULATION MODELS

In order to make the circulation model more generally applicable to prediction of field conditions, both versions of the model were calibrated using field data. The calibrations consisted of determining a monochromatic deepwater wave condition and a uniform wind condition, and then running the models using the field conditions and measured bathymetry. Adjustable coefficients were then tuned to obtain the best possible qualitative and quantitative fit between the currents predicted by the model and those measured in the field.

In this chapter, the field data set used for calibration is described. Results for the circulation models are then shown for a range of coefficient values.

1. FIELD DATA USED FOR CALIBRATION

Field data used during calibration of the nearshore circulation models was obtained from the results of the Nearshore Sediment Transport Study (NSTS) experiment conducted at Torrey Pines beach, near La Jolla, Ca., (see Figure 4-1) in November - December 1978. The results of this study were chosen as being applicable to calibration of the models for several reasons. Torrey Pines beach is a long, straight beach with fairly uniform and parallel offshore contours. The field bathymetry was thus easily adapted into the model's scheme of longshore periodicity.

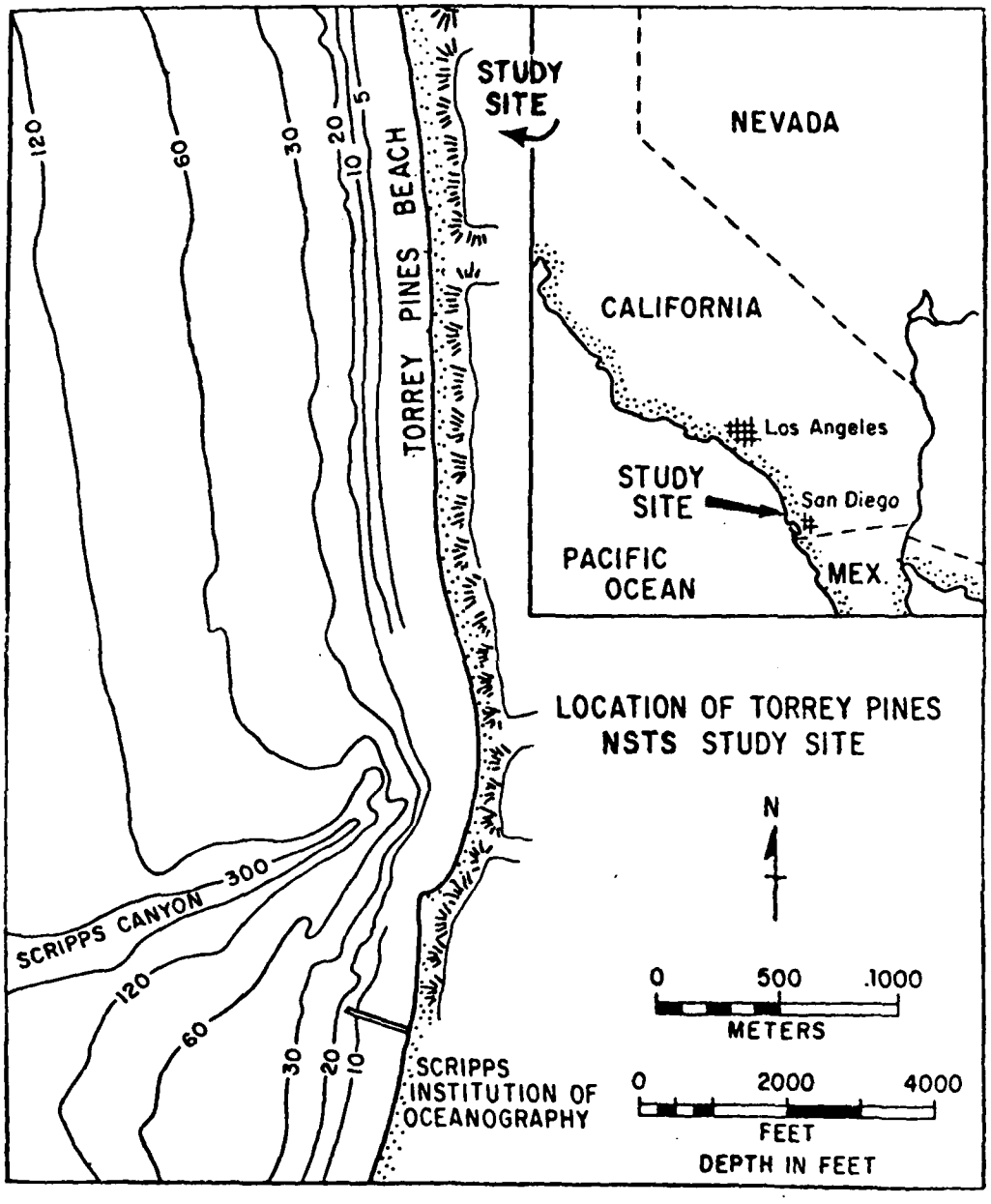


Figure 4-1. Location of NSTS Torrey Pines Experiment. (From Gabler, 1979)

The experiment itself produced a detailed picture of nearshore currents and waves, with up to 22 current meters, 10 pressure gages, and 7 wave staffs being operated simultaneously. Thus the resolution needed to accurately fit the model predictions was present in the field data. Gable (1979) gives a detailed description of the site, instrumentation, and conduct of the NSTS experiment. The arrangement of fixed instruments used in the experiment is shown in Figure 4-2. Angles and distances used in this report will be with respect to the baseline (0 offshore distance) in Figure 4-2. Two separate bathymetry maps are shown in Figures 4-3 and 4-4. It is noted that the survey of Nov. 9, 1978 indicates the presence of a shallow channel oriented almost perpendicular to the baseline at about 40 m left of the main range (0 m alongshore). This feature is not present in the Nov. 18, 1978 survey, which, on the whole, exhibits greater random fluctuation in the location of the contours. Both the channel in the Nov. 9 bathymetry data and the unevenness in the Nov. 18 data appear to be transient features, as will be discussed below.

1.1 Choice of Field Data for Calibration

Several requirements were chosen in order to determine a valid set of field data for comparison to the numerical model. First, the numerical models in their basic state are designed to be run with a monochromatic deepwater wave condition. It was therefore required that the wave field for the chosen data be narrow banded both in a frequency and directional sense. The presence of a second wave component at a different direction than the primary component introduces a forcing mechanism

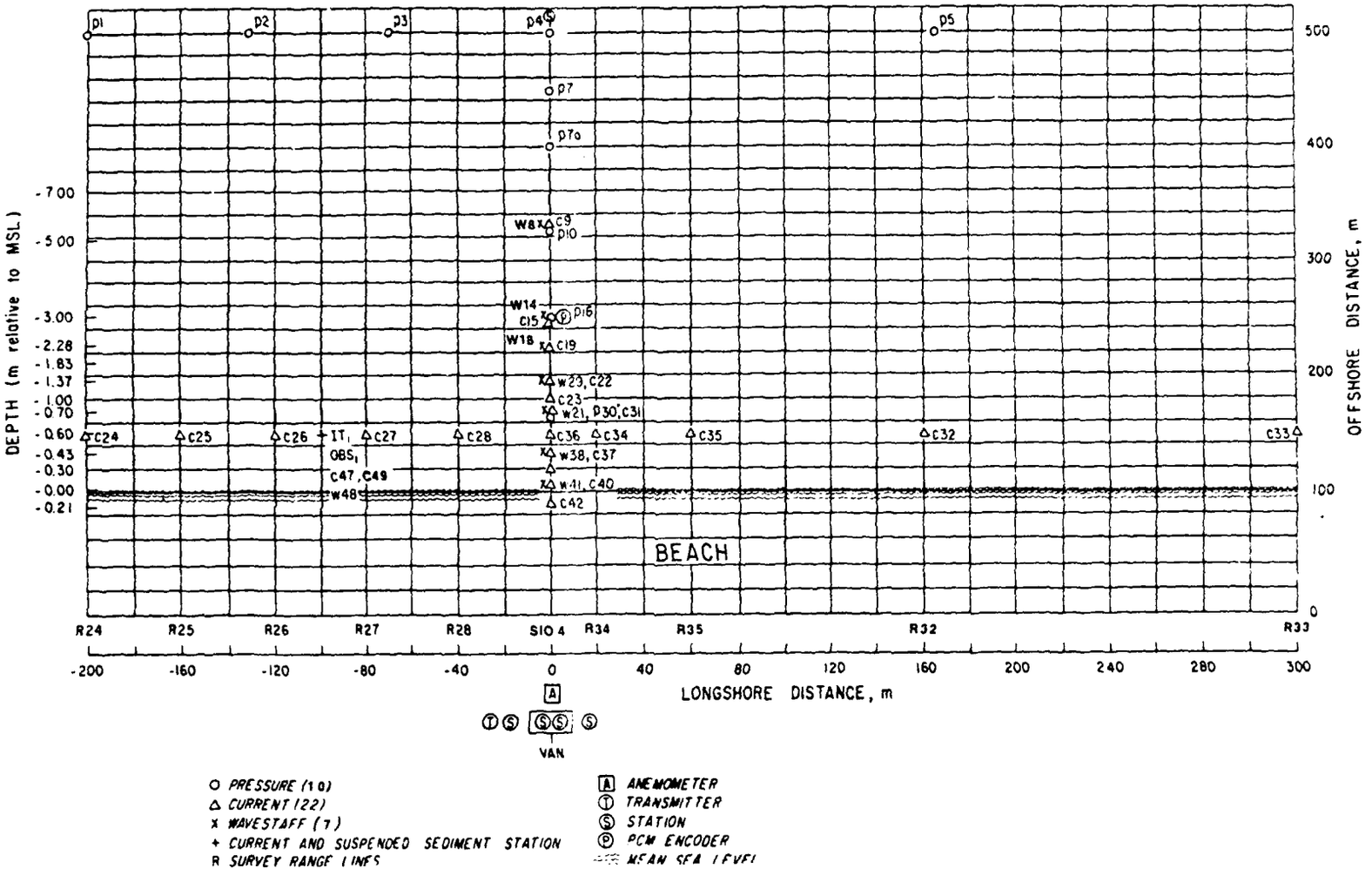


Figure 4-2. Instrumentation for NSTS Torrey Pines Experiment. (From Seibel, 1979)

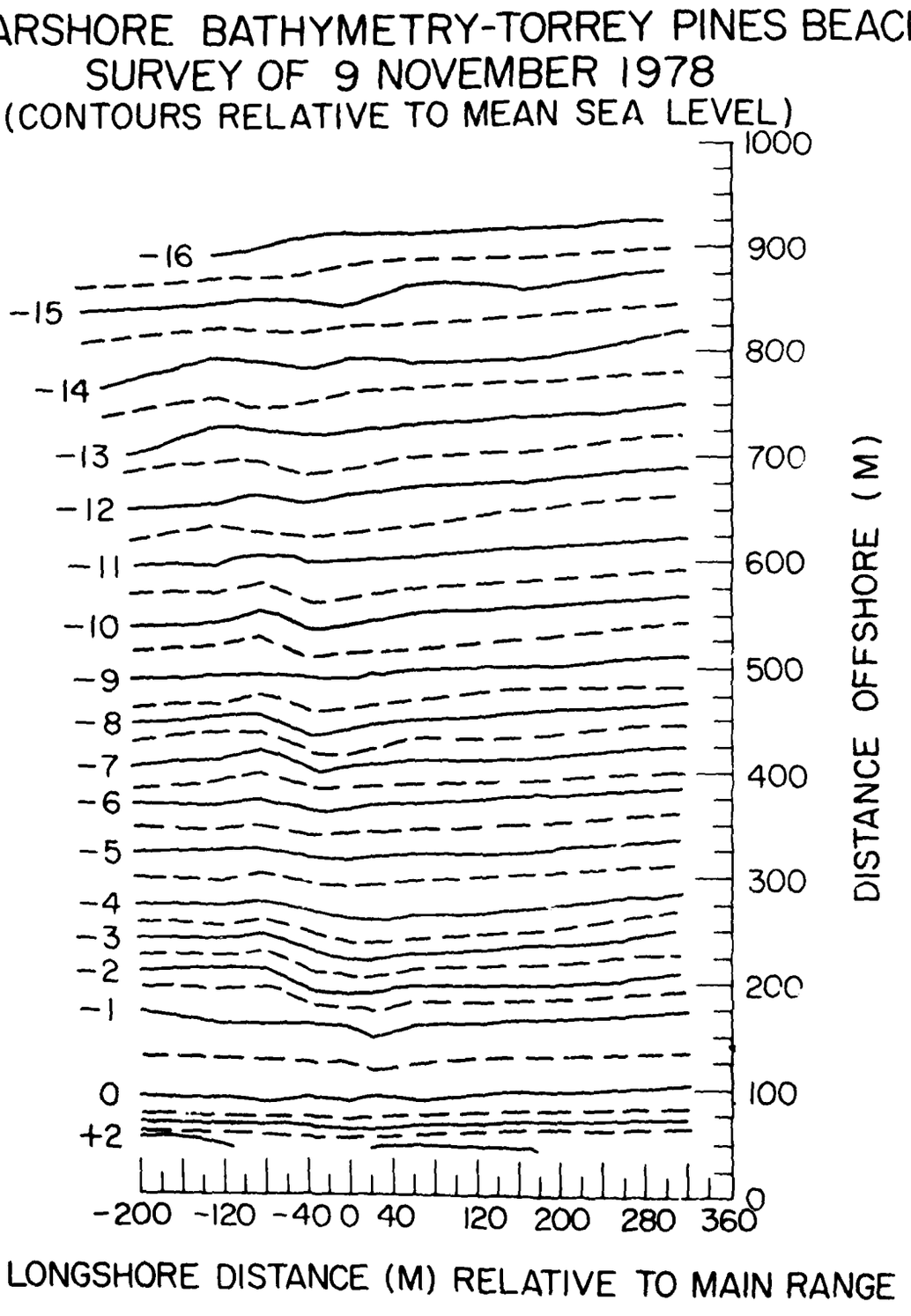


Figure 4-3. Nearshore Bathymetry, Nov. 9, 1978
(from Gable 1979)

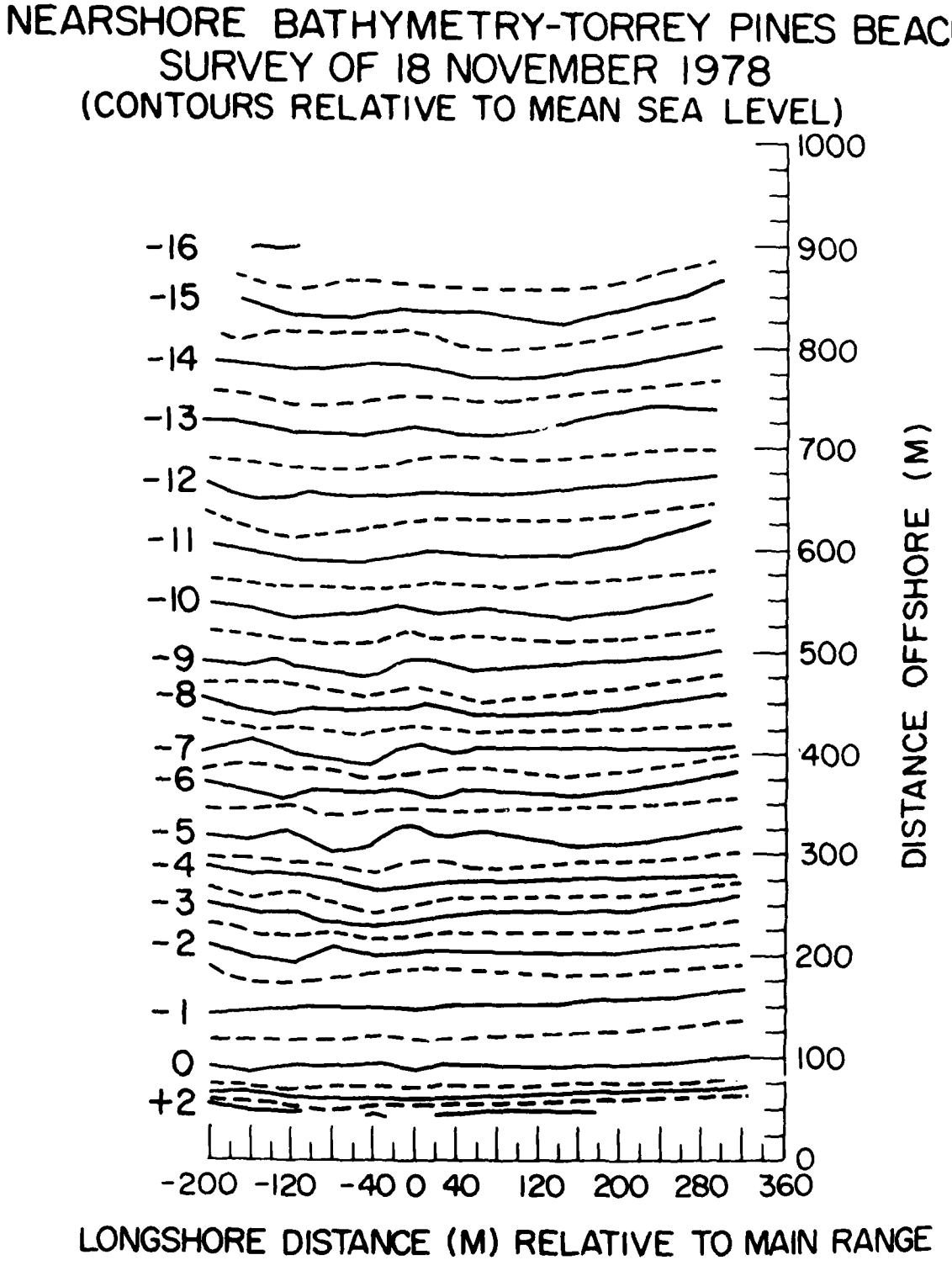


Figure 4-4. Nearshore Bathymetry, Nov. 18, 1978
(from Gable 1979)

which would tend to produce rip cells on a plane beach, as shown by Dalrymple (1975). Just such a case was treated by a simplified refraction model in Ebersole and Dalrymple (1979), but the capability to handle the condition has not been retained in the final model versions since the refraction scheme used does not model the interaction of different waves. In addition, the use of a 17 minute average of the field measured currents in order to obtain fairly steady conditions precluded the modelling of essentially transient circulation phenomenon.

Secondly, since it was desired to calculate a steady state wave and current field, it was required that wave and wind conditions be nearly steady over the interval of averaged field measurements. This required either a quiet or unidirectional wind field as well as fairly steady wave direction, height and period.

The field data set available from the NSTS results was roughly screened on the basis of the beach observations given in Gable (1979). In particular, observation of long crested waves indicated the presence of narrow banded directional spectra. It was found that the NSTS data presented a problem, in that almost all data sets exhibited at least some degree of bi-directionality, with waves approaching from both the north and south. The data chosen for initial calibration has a somewhat smaller wave component from the north than other records, but is still suspect as a valid calibration standard.

Data Set Number 1

The NSTS data tapes divide the current and pressure records into 17 minute segments. The first data set chosen for use in cailbration,

and subsequently used for the majority of calibration runs, was taken from the second 17 minute segment of the test of Nov. 10, 1978. The 17 minute segments were further subdivided into four 4.25 minute segments, and averaged current fields were plotted for each 4.25 minute segment. Two representative plots for the first data set are shown in Figure 4-5a and b.

Based on an average of the 4.25 minute average velocity profiles, an average 17 minute velocity profile was constructed for data values on the main range (offshore array of meters) and is shown in Figure 4-6. The resulting velocity profile indicates a longshore current of about 8 cm/sec magnitude offshore of the surf zone. This current is too far outside the breaker line to be surf driven, and may be due to the presence of a tidal or seasonal current. The effect of this current on the model results is discussed below.

The monochromatic deepwater wave conditions determined for data set No. 1 were as follows:

Wave period	T	:	14.52 seconds
Wave height	H_o	:	0.62 meters
Wave angle	A	:	165.7^o measured clockwise from +x (offshore direction).

The uniform wind conditions were:

Wind speed	W	:	7.2 meters/second
Wind angle	α	:	294.5^o measured clockwise from -x (onshore) direction.

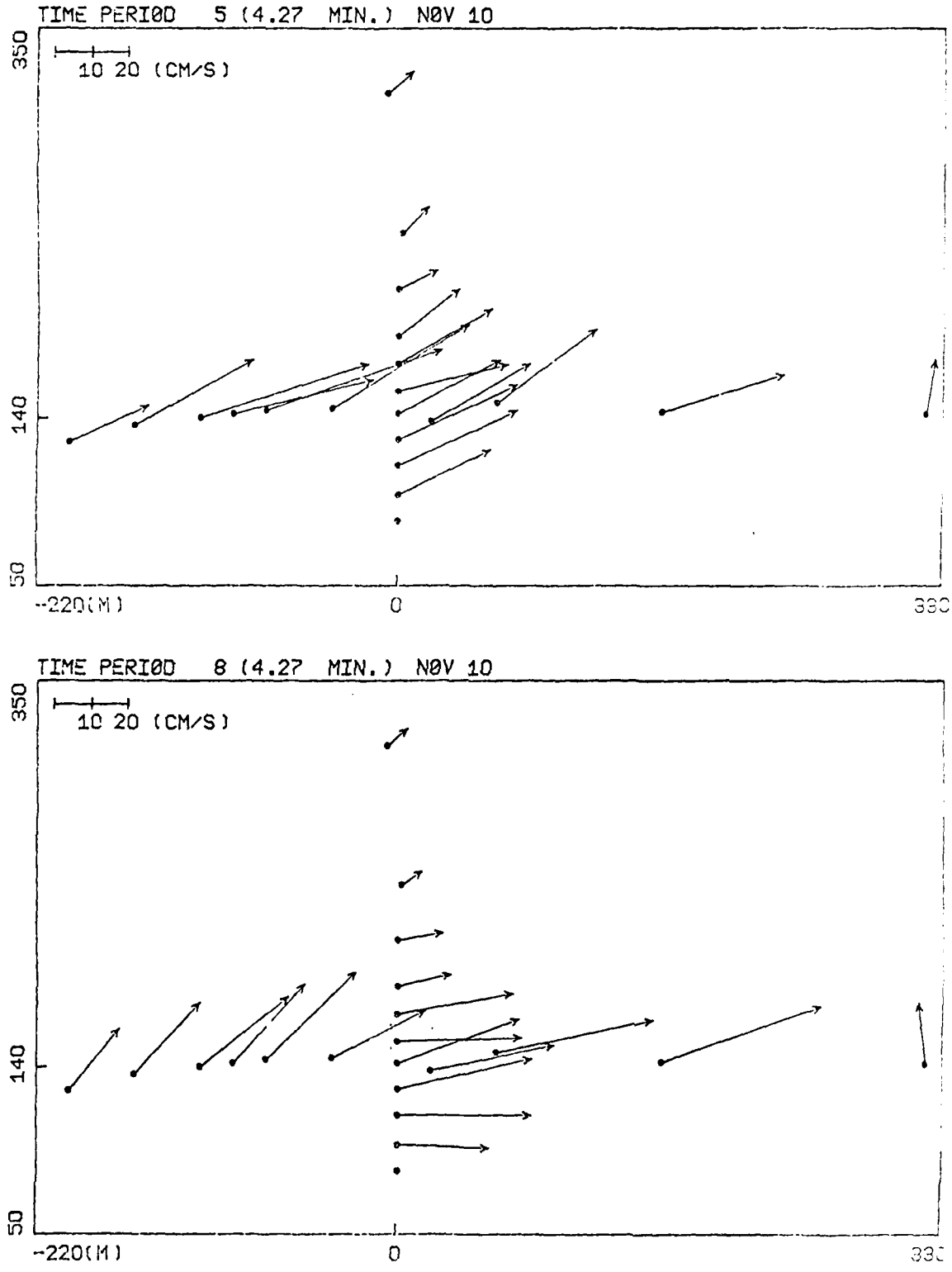


Figure 4-5. 4.27 Minute Average Current Vectors, Nov. 10, 1978.

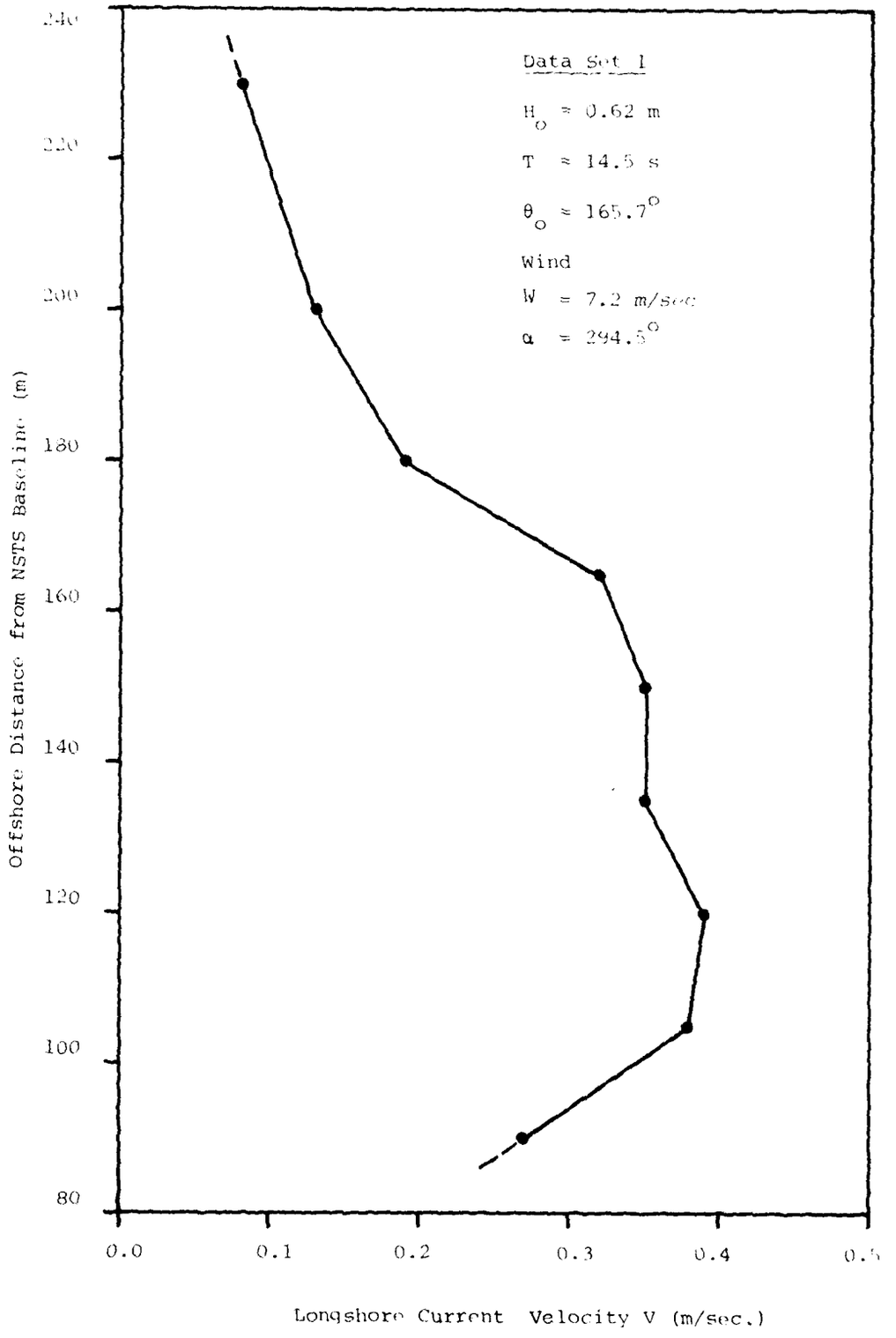


Figure 4.6 Average Field Velocity Profile Data Set 1.

Data Set Number 2

Data Set Number 1 represented the only strongly unidirectional, narrow banded wave spectrum contained in the Torrey Pines data. The remaining data sets, even in the absence of locally generated short wind waves, exhibited at least a strong tendency towards bidirectionality, with spectral peaks nearly symmetrically placed about the shore-normal axis. Since it is anticipated that measured wave fields in general would seldom exhibit the narrow banded, unidirectionality required as input into the model, a data set was constructed based on averaging over the parameters of the measured wave field in order to test the response of the model using a "best guess" for the desired input parameters.

The data chosen for this test was from the eighth 17-minute time period of the November 15 data. Wave energy was contained in two directional peaks in a narrow frequency band (Figure 4-7). Using directional spectra supplied by Pawka (1980), the data set was constructed by summing the mean energy density over all frequencies in the dominant spectral peak. The wave angle was chosen as the average of the dominant angle at each frequency weighted by the energy densities. The peak energy containing frequency was taken as the dominant frequency. The resulting data set follows:

Wave Period	T	:	14.71 seconds
Wave Height	H_o	:	0.43 meters
Wave Angle	A	:	168° measured clockwise from +x (offshore direction)

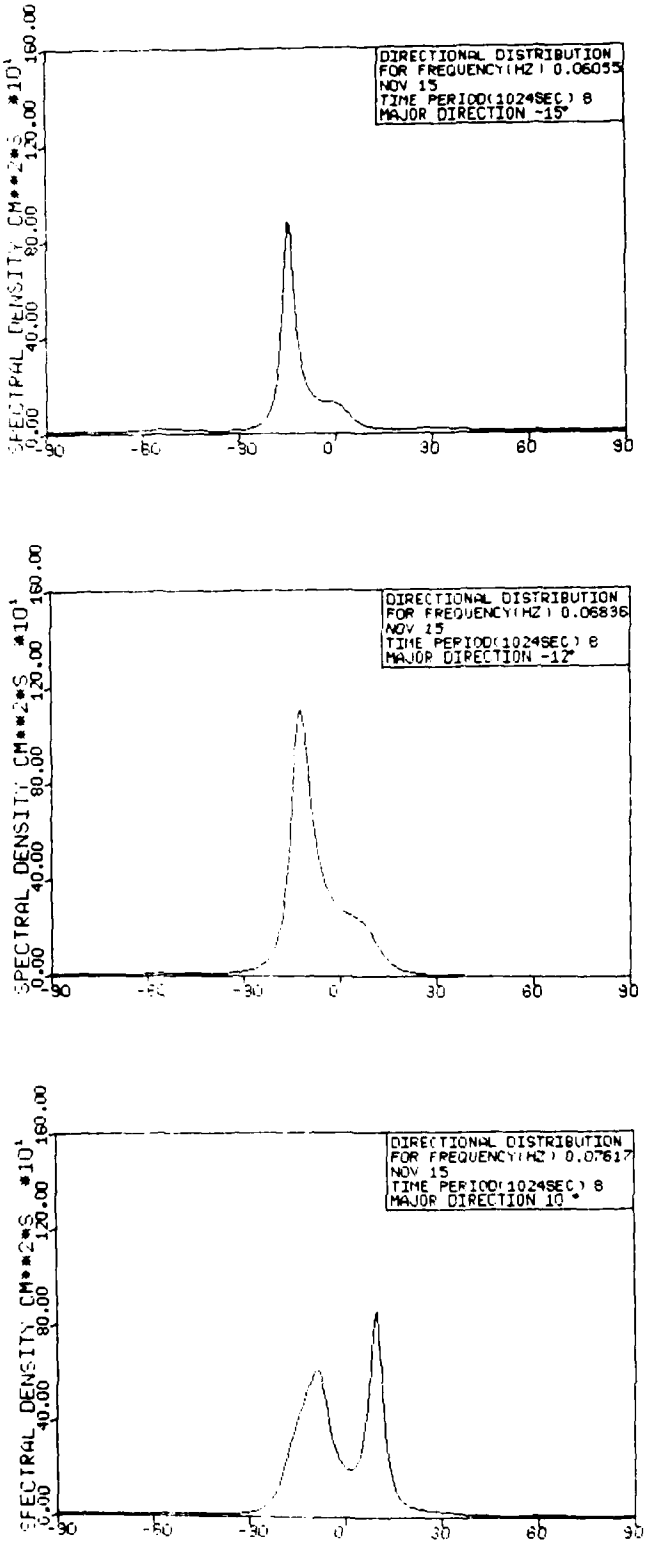


Figure 4-7. Directional Distribution of Wave Energy at Peak Frequencies, Nov. 15, 1978 (supplied by S. Pawka, 1980)

There was no significant steady wind component.

Wind speed W : 0.0 meters/second

Wind angle α : not applicable.

The resulting data set has an indeterminant connection with the measured field data, and at best the model could be expected to exhibit results in qualitative agreement with the field data.

Figure 4-8 shows a plot of average longshore current over the 17 minute period of data collection. Individual plots (not shown) of 4.25 minute averages show a great deal of scatter in peak velocities and shape of the profile, indicating some unsteadiness in the current field, as would be expected due to the bi-directional wave field.

2. CALIBRATION OF THE MODELS

Both the linear and nonlinear models have a bottom friction factor f and the Van Dorn coefficients K_1 and K_2 as adjustable parameters. In addition, the nonlinear model has adjustable coefficients of lateral eddy mixing in both the longshore and offshore directions.

The Van Dorn wind stress formulation used in the models is intended to represent the transfer of momentum from a wind field to the water column, leading to a wind-induced longshore current for wind stress applied in the long shore direction, and wind set up for wind blowing towards shore. Wind set up cannot currently be accurately calibrated in the present models due to the artificial barrier at the offshore grid row; no additional water can

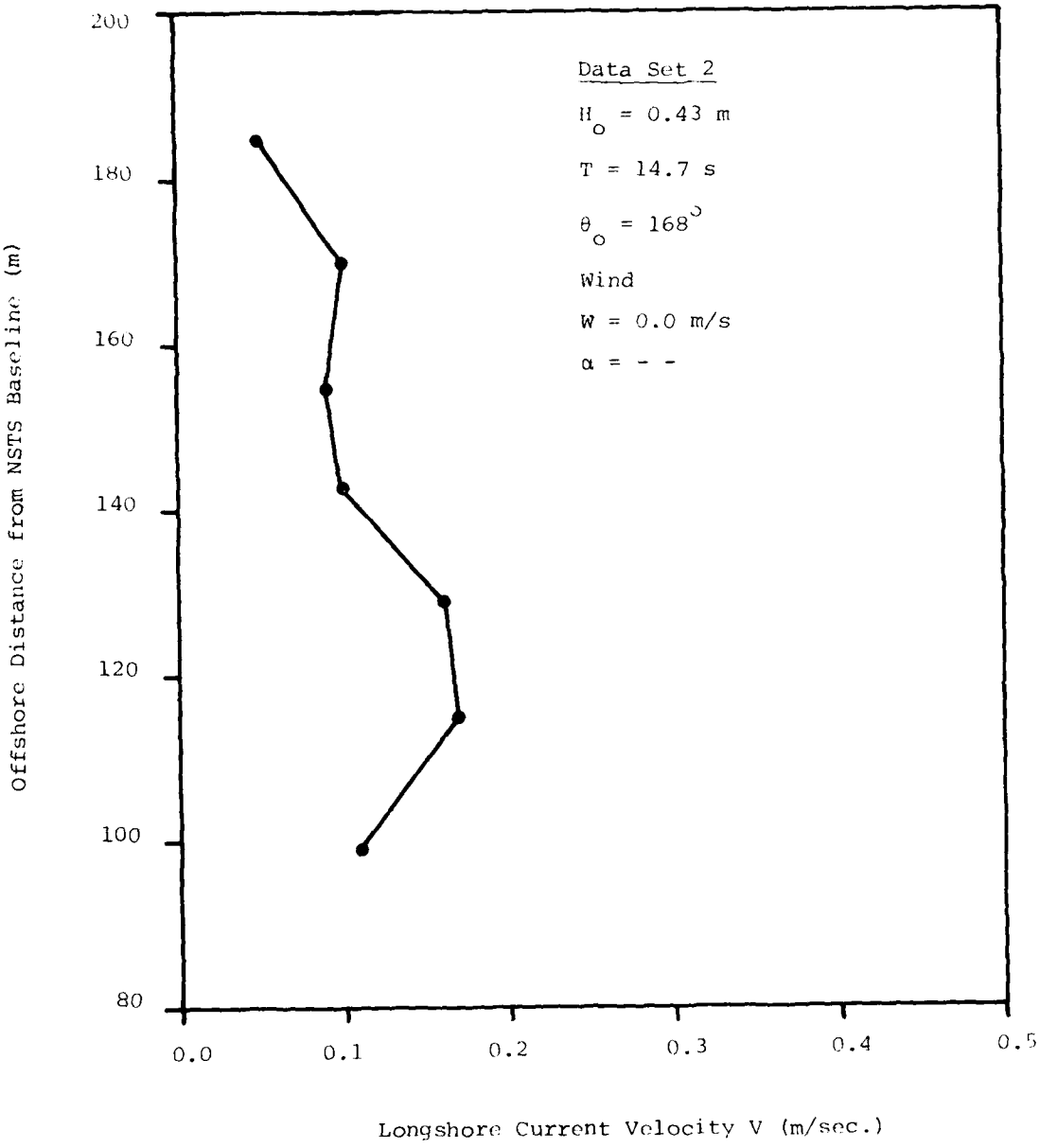


Figure 4-8 Average Field Velocity Profile Data Set 2

enter the model after the start of the run with a given depth grid. The parameters used in the Van Dorn formula have been tested in previous large scale models (Pearce, 1972), and have been found to be satisfactory.

With the elimination of the wind stress coefficients, only the bottom friction coefficient, f, remained to be calibrated in the linear model. The procedure used was to choose a value of f for both models based on a comparison of field data with the linear model. The value of f chosen was then used in the nonlinear model as a first approximation, and values of the eddy mixing coefficients were adjusted to again obtain a best fit between the results of the nonlinear model and the field data. It should be noted, however, that there is no a priori reason that both models should behave optimally with the same value of f.

2.1 Linear Model Calibration

Runs of the linear model were conducted using the deepwater wave conditions of Data Set 1 and the measured bathymetry of Nov. 9, 1978. For all values of f chosen, rip currents formed near the shallow channel seen in the Nov. 9 bathymetry. These rip currents were not apparent in the Nov. 10 currents (Figure 4-5), where they would affect the main range velocity profile. An example of the current field calculated using the Nov. 9 bathymetry is shown in Figure (4-9). The value of f for the run shown was 0.01.

It was tentatively concluded at this point that the shallow channel seen in the Nov. 9 data was a transient feature and was not present on Nov. 10, juding from the uniformity of the measured currents. It was decided to run a model using a bathymetry based on a longshore average of the Nov. 9 bathymetry

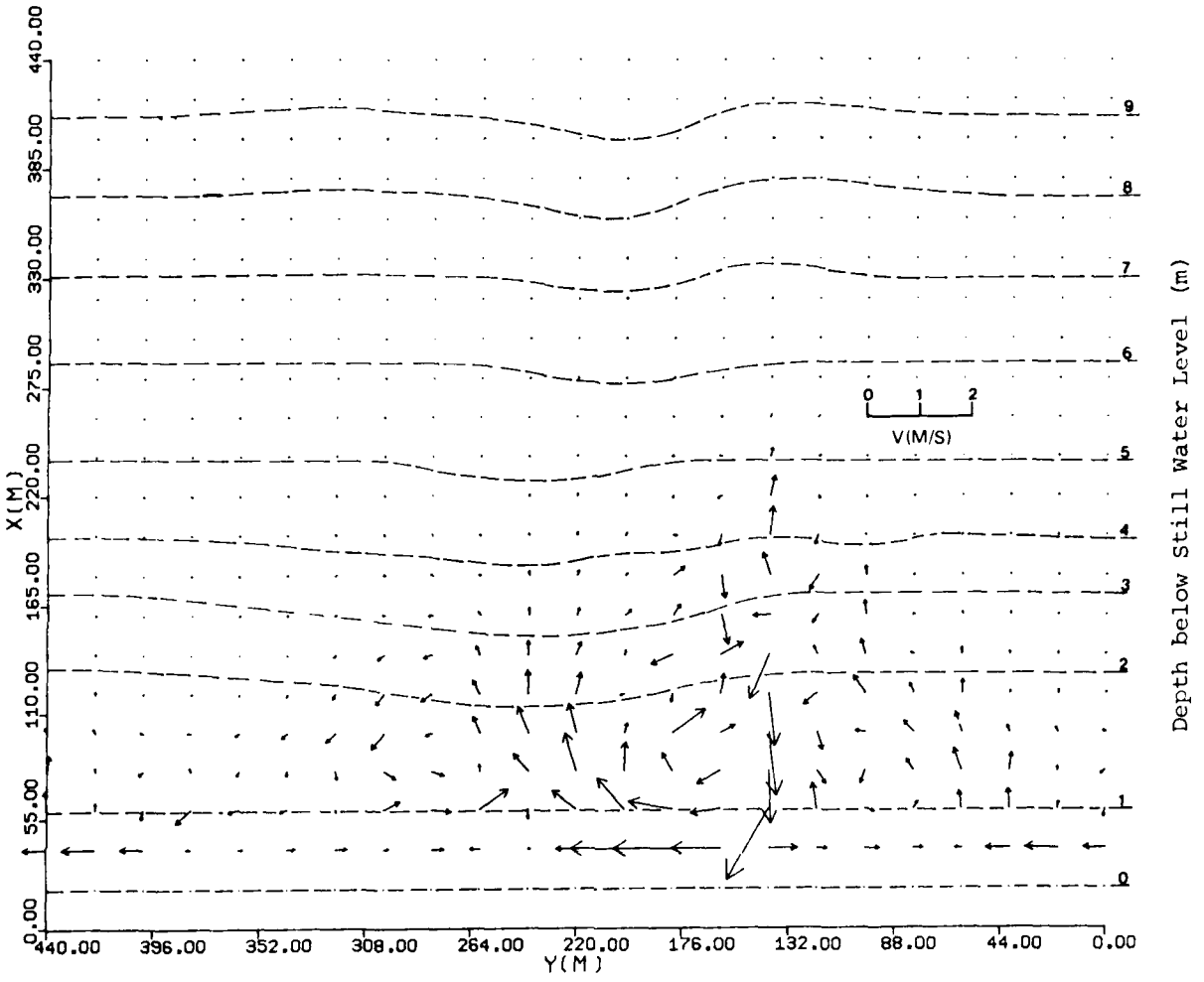


Figure 4-9 Currents Induced in Field Using Nov. 9 Bathymetry Nonlinear Model: Data Set 1.

data. The longshore extent of the beach was arbitrarily chosen as 200 meters. The profile is shown in Figure 4-10.

Using the longshore averaged bathymetry, model runs were conducted for a range of f values. Velocity profiles for values of f of 0.01, 0.015 and 0.02 are shown in Figure 4-11, in comparison to the current distribution measured in the field. In addition, a "corrected" field current distribution constructed by subtracting the 0.08 m/sec offshore current is shown as the dashed line.

Figure 4-11 shows that, by using a value of f equal to 0.015, the linear model closely predicts the maximum velocities and the general velocity distribution in the surf zone. The longshore current predicted by the model dies off more quickly offshore due to the absence of lateral mixing effects in the linear model.

2.2 Nonlinear Model Calibration

The nonlinear model was run using Data Set 1 and the value of f = 0.015 obtained from the linear model calibration. Three sets of mixing coefficients ε_x (or N) and ε_y were used (see Equation (2.40)).

N	ε_y	
0.000	0.00	no mixing
0.0025	0.25	
0.005	0.50	

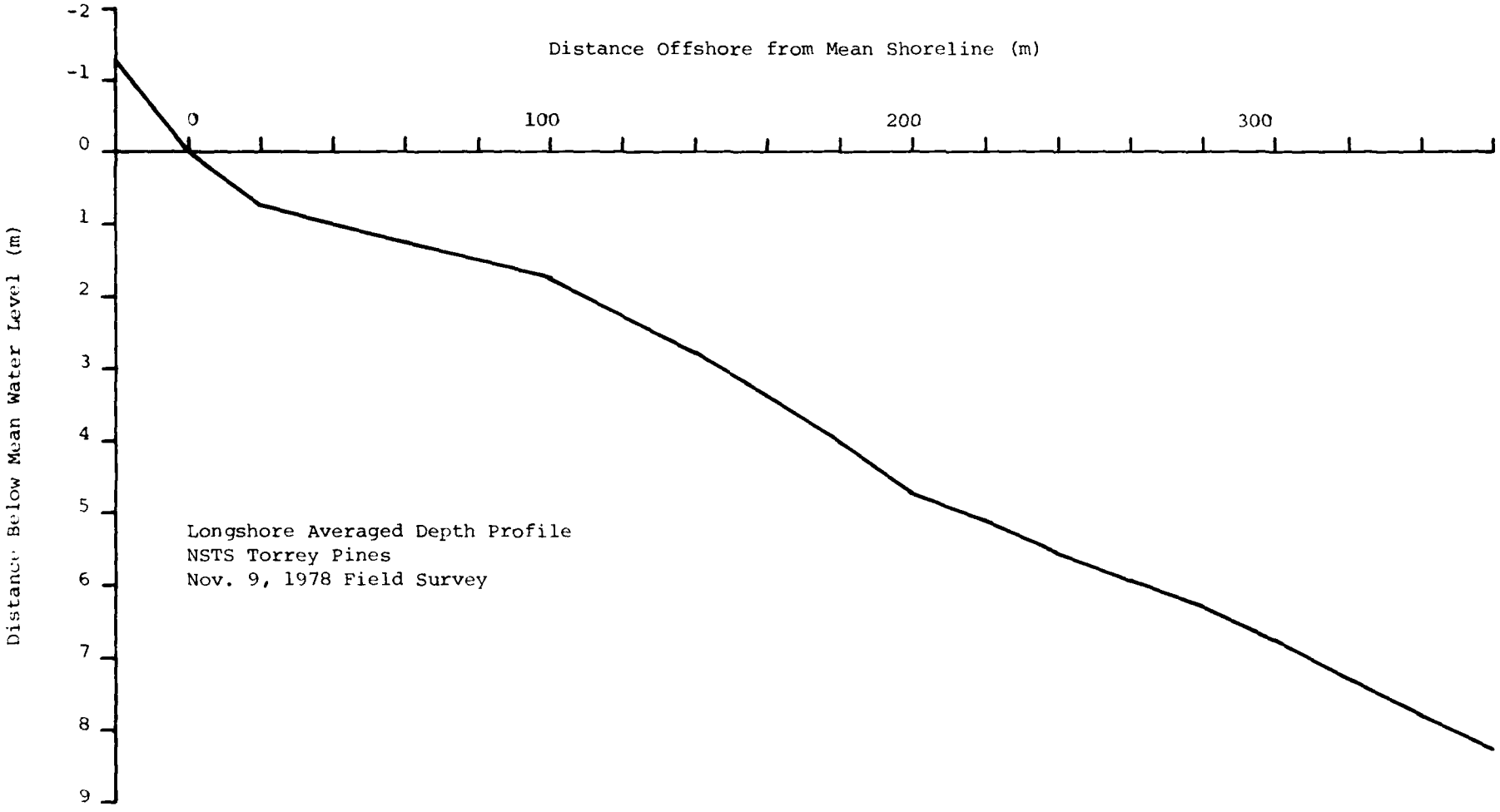


Figure 4-19 Longshore Average Depth Profile, Torrey Pines, November 9, 1978.

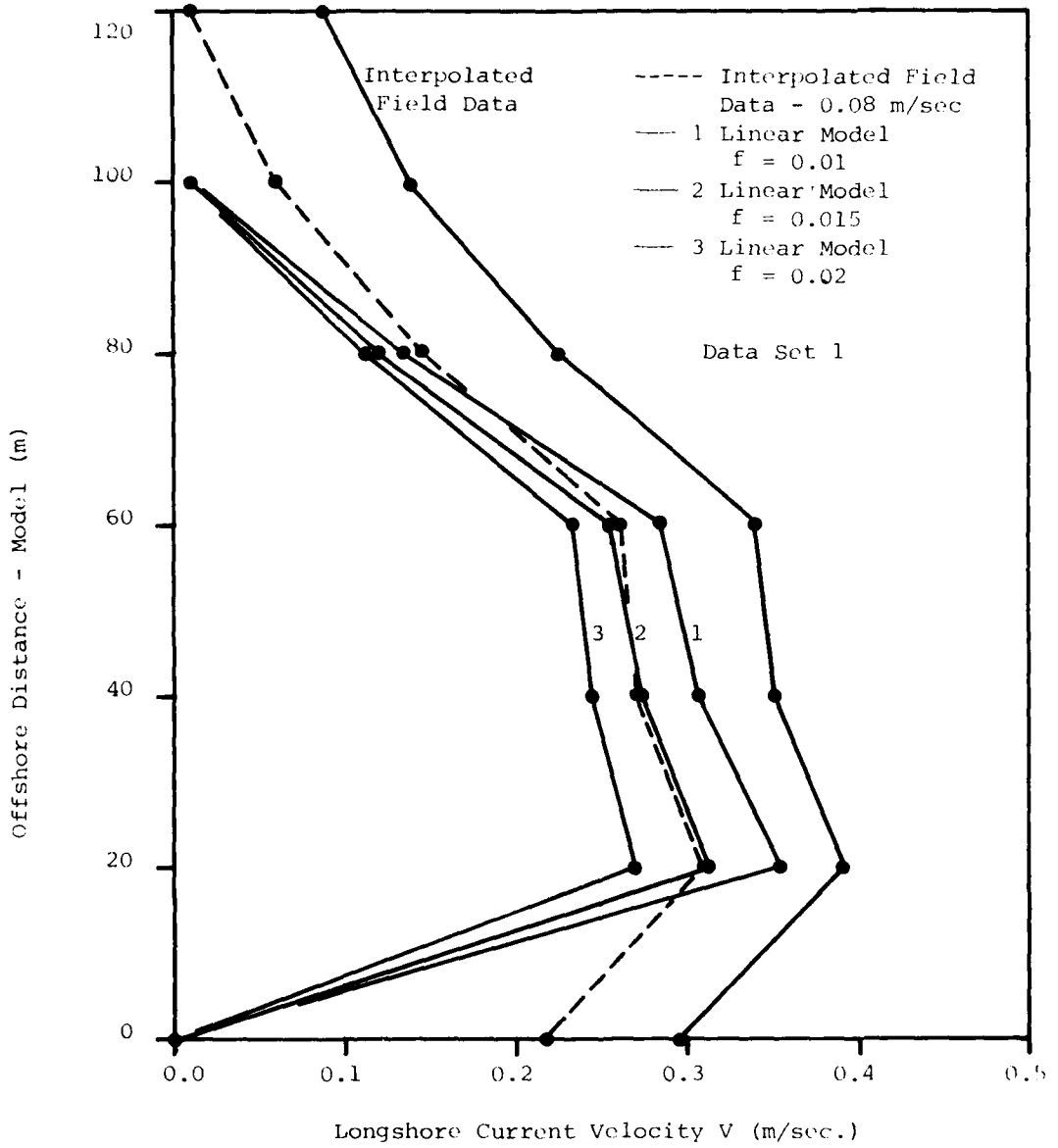


Figure 4-11 Variation of Longshore Current with Friction Factor in the Linear Model.

Results for f = 0.015 are shown in Figure 4-12. It was found that the inclusion of mixing satisfactorily extended the velocity profile in the offshore direction, but in both cases, including mixing, the current magnitudes in the surf zone were underpredicted. The model was therefore retested using a smaller value for the friction coefficient, f = 0.01. Results are shown in Figure 4-13. The smaller value of f is seen to correct for the underprediction of longshore current. Coefficient values for the nonlinear model were chosed based on the second set of results,

$$f = 0.01$$

$$N = 0.0025$$

$$\varepsilon_y = 0.25 \quad .$$

2.3 Response of Both Models Using Data Set 2

The linear and nonlinear models were run using Data Set 2 and the calibrated coefficients obtained above. Results for the linear model are shown in Figure 4-14. Nonlinear model results were similar to the linear model results. Both models were seen to overpredict the maximum longshore current in comparison to the averaged field data, and to underpredict the offshore extent of the longshore current. It is likely that a better fit to the field data could be obtained by artificially increasing the lateral mixing in the nonlinear model. However, the field data represents the average of a complex flow pattern, where the effect of mixing in the averaged data is induced by large scale phenomenon not likely to be found in the steady current fields induced by a unidirectional wave field. Since it is likely that the magnitude of artificial mixing would vary greatly as a function of

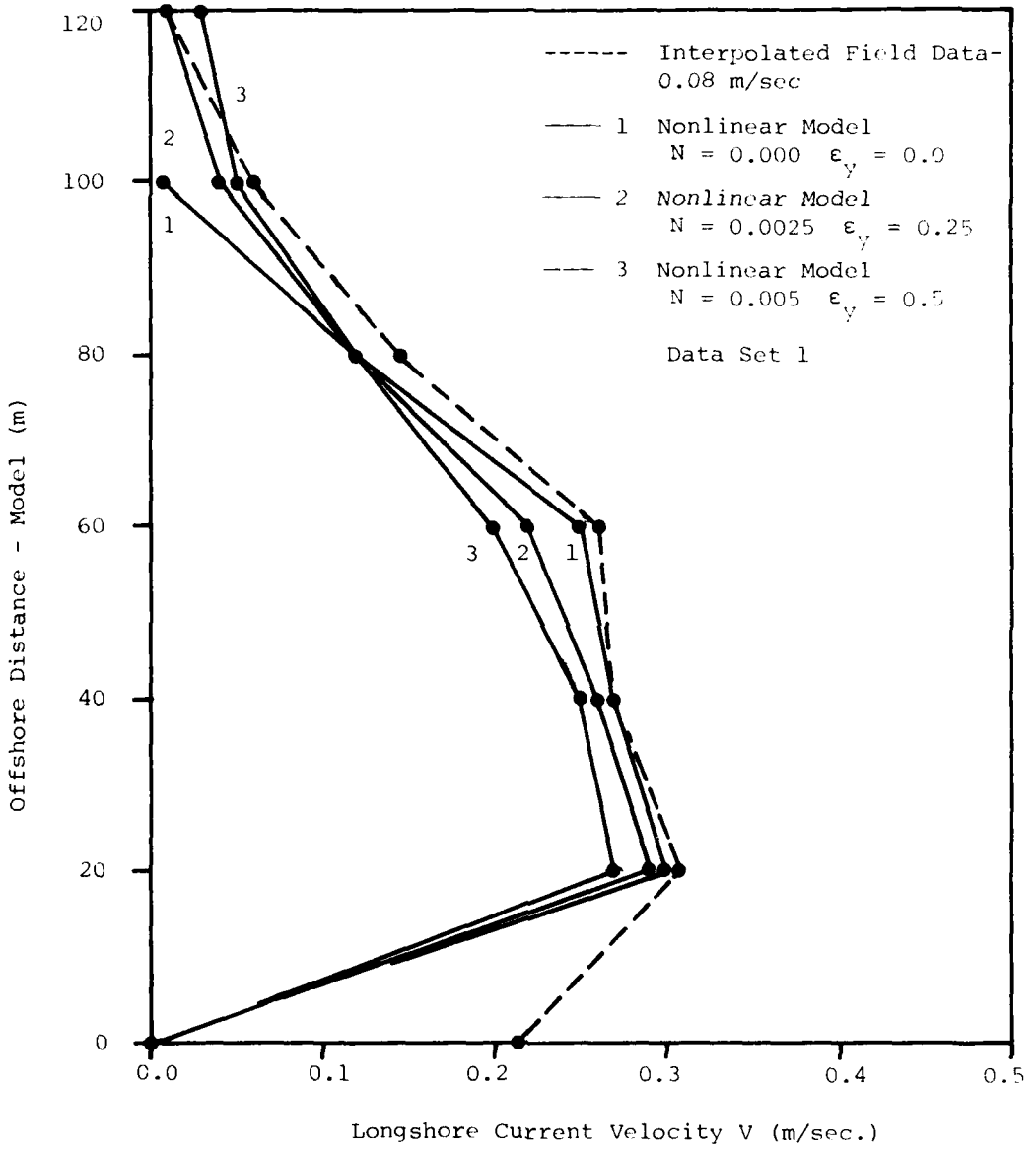


Figure 4-12 Variation of Longshore Current with Lateral Mixing in the Nonlinear Model.
f = 0.015

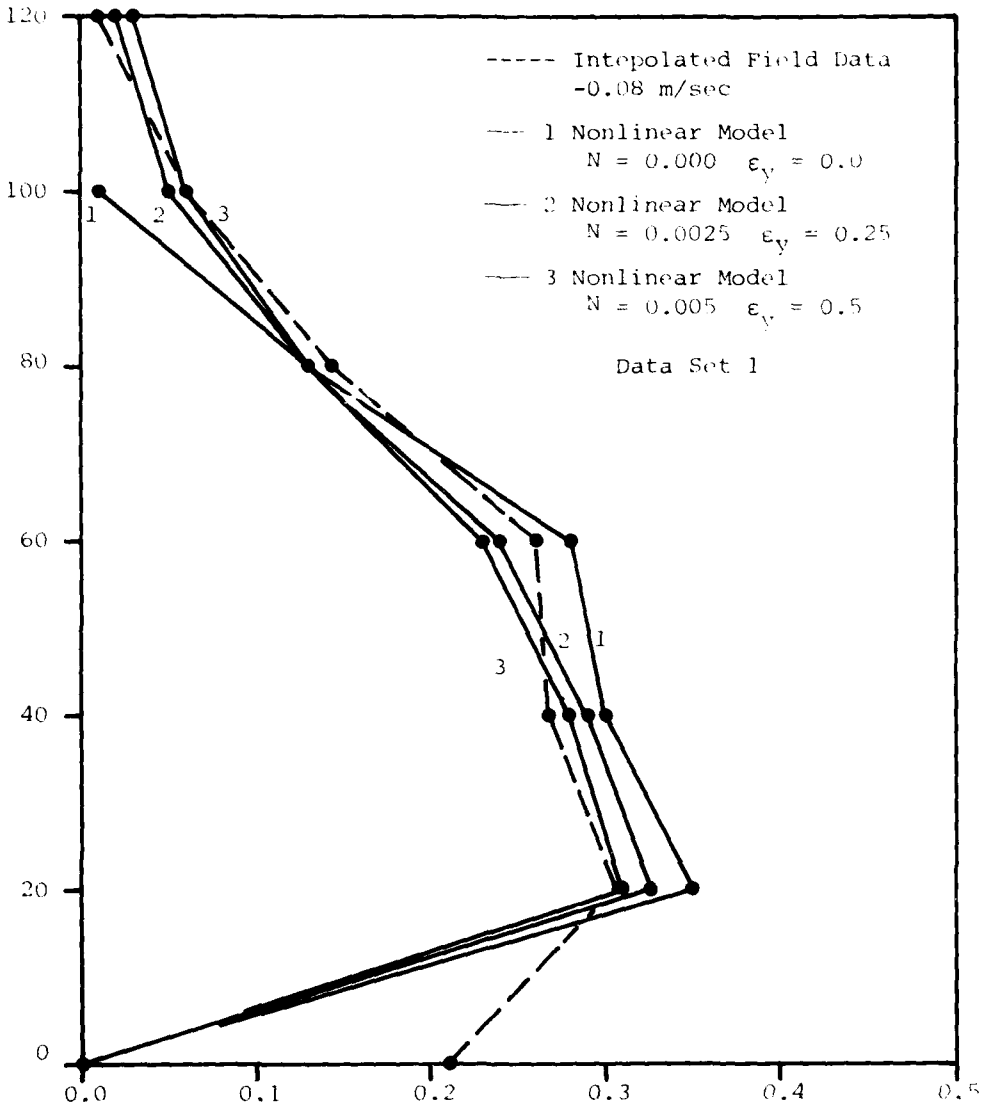


Figure 4-1[illegible] Variation of Longshore Current with Lateral Mixing in the Nonlinear Model.
f = 0.010

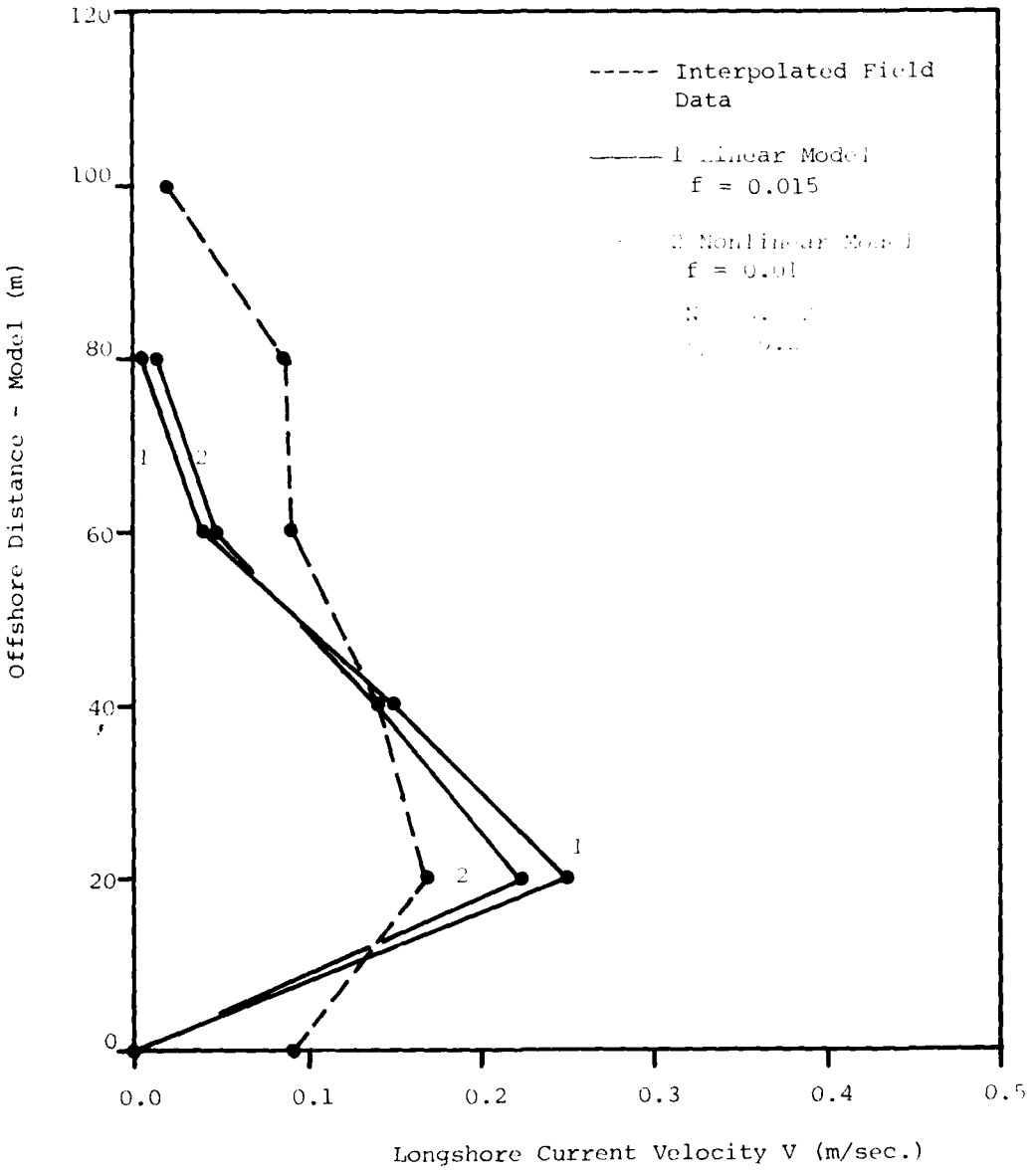


Figure 4-14 Longshore Current in the Linear and Nonlinear Models. Data Set 2

wave directionality and spatial complexity, it would be unjustified to alter the parameters of the model to fit a single case. It may become possible at some future date to estimate mixing parameters based on the characteristics of the random wave field.

A plot of the velocity vectors for data set 2 obtained using the calibrated model and the November 18 bathymetry is shown in Figure 4-15. As in the November 10 case, the circulation pattern is seen to be sensitive to the bottom variations. This result is probably due to the low values obtained for the friction coefficient f.

3. DISCUSSION OF THE CALIBRATED COEFFICIENTS

The value of the friction factor f obtained in this study differs significantly from the value used in previous work and retained by Allender et al. (1981). Based on a bottom shear stress relation given by Eqs. (2.27-2.28), Birkemeier and Dalrymple (1976) chose a value of $C_f = 0.01$, which corresponds to a choice of $f = 0.08$. This value of f is carried over into the results discussed in the next chapter. However, model calibration has indicated that the value of f is more of the order 0.01-0.02, with values of 0.015 and 0.01 chosen for the linear and nonlinear models respectively. It is felt that this alteration in choice of the friction factor requires some discussion.

Many formulae exist to calculate the bottom friction factor under waves, the most successful being those of Jonsson (1966) and Kajiura (1968). Writing Kajiura's formula in a form given by Dalrymple and Lozano (1978), we obtain

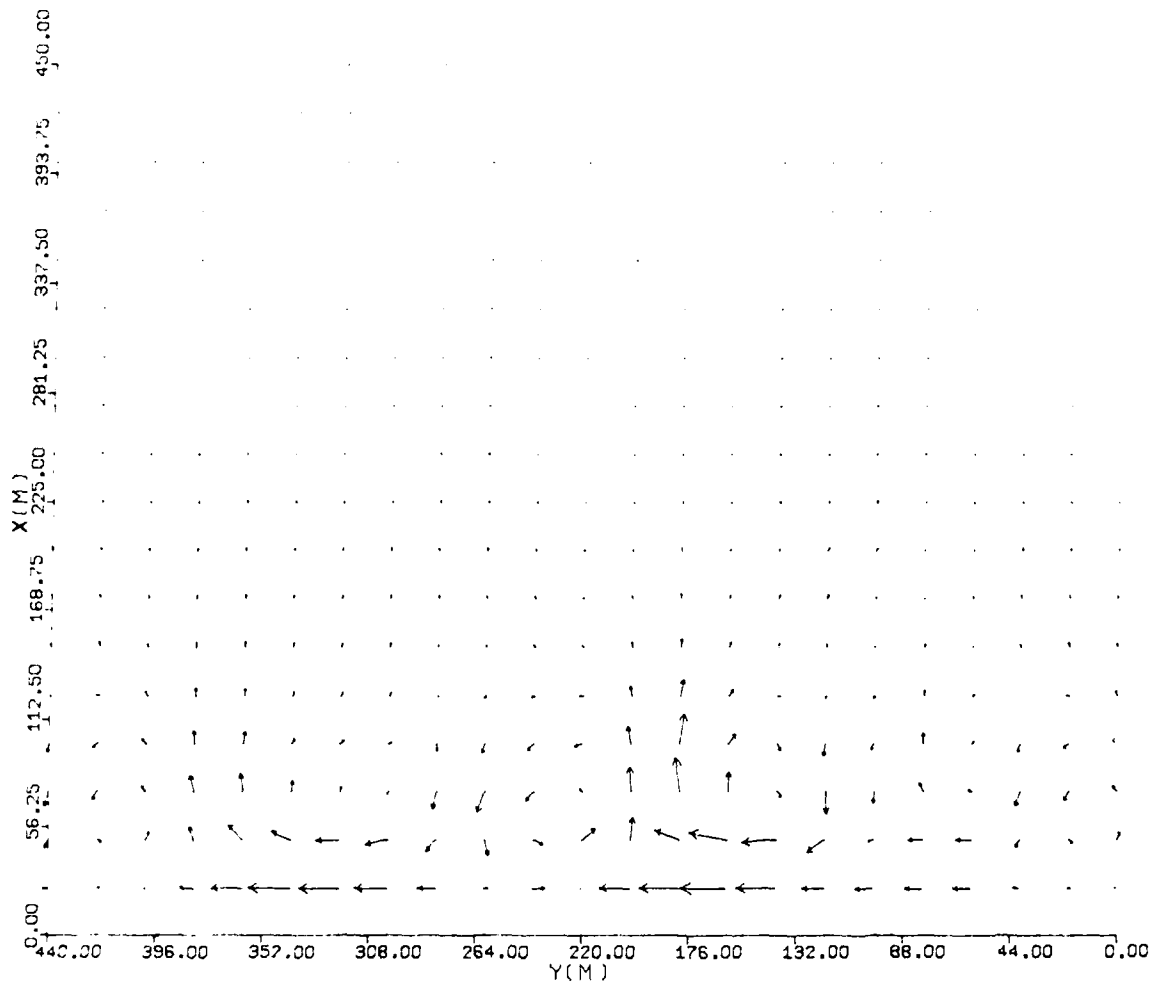


Figure 4-15. Currents Induced on Nov. 18, 1978 Bathymetry Using Data Set 2: Nonlinear Model.

$$C_f = \frac{f}{8} = 1.41 \left\{ \frac{4\pi d}{T(gh_b\kappa^2)^{1/2}} \right\}^{2/3}$$

where d is the median sand grain dimeter and κ is the breaking index. The value of f at the breaker line represents a reasonable choice for an average value over the surf zone region. For Torrey Pines Beach, d is approximately given by

$$d \sim d_{50} = 0.27 \text{ mm}$$

For the field data used, h_b is taken as approximately 1.5 m. This yields a value of

$$C_f = .0025 \text{ or } f = 0.02$$

The values of f obtained during calibration are thus of the correct order of magnitude for flows over a planar bed with no ripples. However, the real physical bottom being studied should exhibit a higher roughness, with length scales based on ripple geometry rather than the sand grain diameter, indicating that the initially chosen value of $C_f = 0.01$ is probably more correct on physical grounds.

In this regard, we note the questionable practice of using distinctly bi-directional wave data to generate monochromatic input wave conditions for calibration purposes. The calibration obtained here possible constitutes a valid site specific calibration for the Torrey Pines beach, since it was found to be possible to predict net longshore flows resulting from wave fields with several components. In a practical sense, few data sets exist which are strictly useful for calibration purposes.

The values of the offshore mixing coefficient N chosen in this study are approximately one order of magnitude smaller than the corresponding value suggested by Bowen and Inman (1974), who investigated a group of dye dispersion studies performed by various investigators. Some indication that a larger value of N may be desirable is given by Figure (4-9): however, the validity of the field bathymetry is suspect in this case. It should be noted that the model exhibits a certain degree of numerical diffusion when large grid spacings are used, indicating that the value of N chosen should be less than the corresponding physically realistic value in any case.

Chapter V

EXAMPLES OF NUMERICAL RESULTS USING THE NEARSHORE CIRCULATION MODELS

1. INTRODUCTION

Various results of calculations using the linear and nonlinear models have been described in detail in Birkemeier and Dalrymple (1976) and Ebersole and Dalrymple (1979). Results pertaining to the questions of model stability and convergence have been mentioned in the previous chapters, with special attention to the seiching mode of oscillation generated at model start-up. In this chapter, results of model calculations in specific situations will be discussed, with emphasis on comparison to analytic models, and to bottom topographies which reproduce earlier efforts.

In addition to the general facilities for user-defined input, each of the models described in the two previous studies contained specialized facilities for the purpose of illustrating specific situations not covered by the basic model structures. These special cases are discussed here for completeness, although in most cases the final model versions may not retain the corresponding capability.

2. SPECIFIC APPLICATIONS OF THE LINEAR AND NONLINEAR MODELS

In this section, we review results presented by Birkemeier and Dalrymple (1976) and Ebersole and Dalrymple (1979). In addition, we present results for an additional form of generalized topography.

2.1 Plane Beach Applications

Historically, the first application of the averaged momentum flux formulation presented in Chapter II to nearshore dynamics was made in an attempt to explain the phenomenon of wave set-up and longshore wave-induced currents in the surf zone. In the simplest formulation, the equation lead to a linear longshore current profile as shown in Figure 2-2 (Longuet-Higgins, 1970a). The addition of a turbulent stress term representing the effect of lateral mixing leads to a smoothed profile which is more representative of observed current distributions, as discussed by Longuet-Higgins (1970b). In the development in Chapter II, a lateral mixing model in the x and y directions was outlined based on the model used by Longuet-Higgins. Here, we review results calculated both with and without the lateral mixing effects.

Birkemeier and Dalrymple (1976) presented results for set-up resulting from normally incident waves on a plane slope, as shown in Figure 5-1 in comparison with the latoratory measurements of Bowen et al. (1968). Since, for this case, no currents are generated, the distinction between the linear and nonlinear models are inapplicable. The model is seen to accurately reproduce aspects of the solution based on linear incident waves, including the sharp drop in the mean water level just outside of the breaker line. It is noted that this drop can be eliminated, and a more realistic solution obtained, by the use of radiation stress terms derived from cnoidal wave theory (James 1974)). This possible refinement has not been included in the models.

Birkemeier and Dalrymple also investigated the dynamic set-up resulting from a time varying incident wave height given by

$$H_o = H_s + A \sin \left(2\pi n \frac{\Delta t}{T_g}\right) \tag{5.1}$$

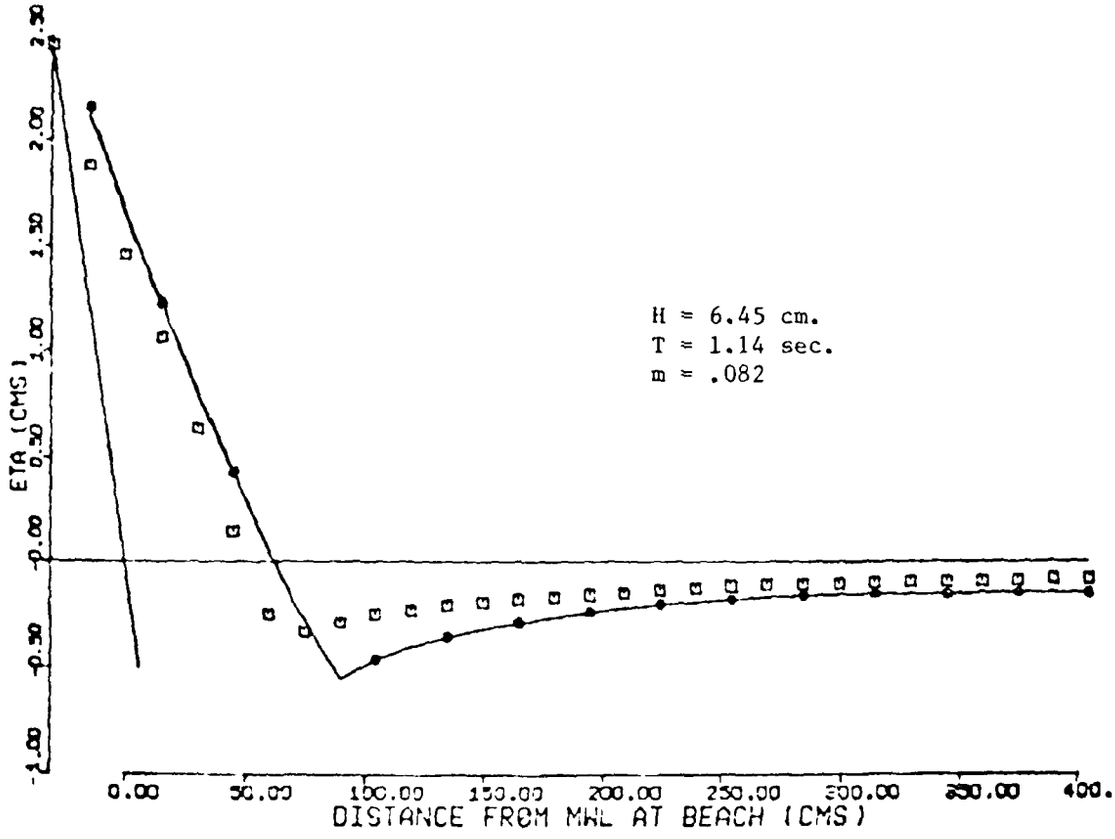


Figure 5-1. Set-up on a Plane Beach

⊡ - Bowen et. al. (1968) experiment
. - Linear Model

where

H_s = starting wave height

A = amplitude of variation

n = iteration index

T_g = group period.

Results for dynamic set-up in comparison to incident wave height are shown in Figure 5-2. As an example of the seiching response of the model, a case was also run where the group period corresponded to the seiche period of the model as given by Eq. (3.5). Figure 5-3 indicates the resulting instability in the set-up which represents the growth of the seiche.

In the previous example, a model seiche appeared as a forced response to the applied wave-induced stress. Figure 5-4 indicates a more typical result of a seiching response to the transient model start-up. In this case, the oscillation is a free motion which gradually dies away due to the effect of bottom friction (as well as numerical dispersion). As mentioned in section 3.4, this type of response can be reduced by a gradual build-up of the incident wave height.

Results for a longshore current on a plane beach are shown in Figure 5-5, for the linear model and no lateral mixing, and in Figure 5-6, for the nonlinear model with mixing. The two solutions clearly represent the behavior predicted by the theoretical solutions, with the exception that the sharp decay in velocity at the breaker line predicted by the solution with no mixing is spread over several grid spaces in the linear model due to the use of finite grid squares. This effect can be reduced by the use of a finer grid spacing.

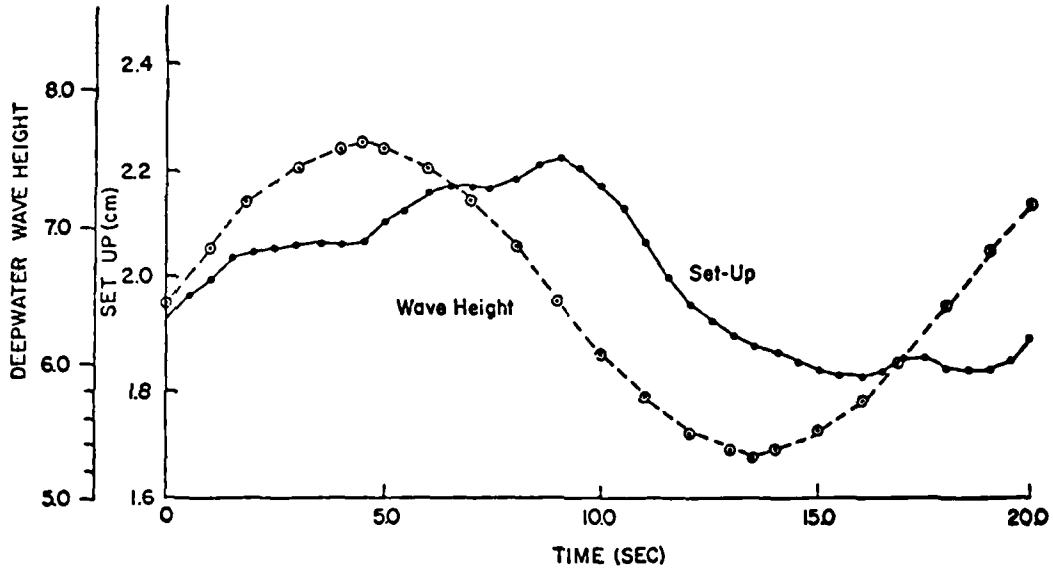


Figure 5-2. Set-up at Shoreline due to a Wave Group with 18 second period.
Linear Model

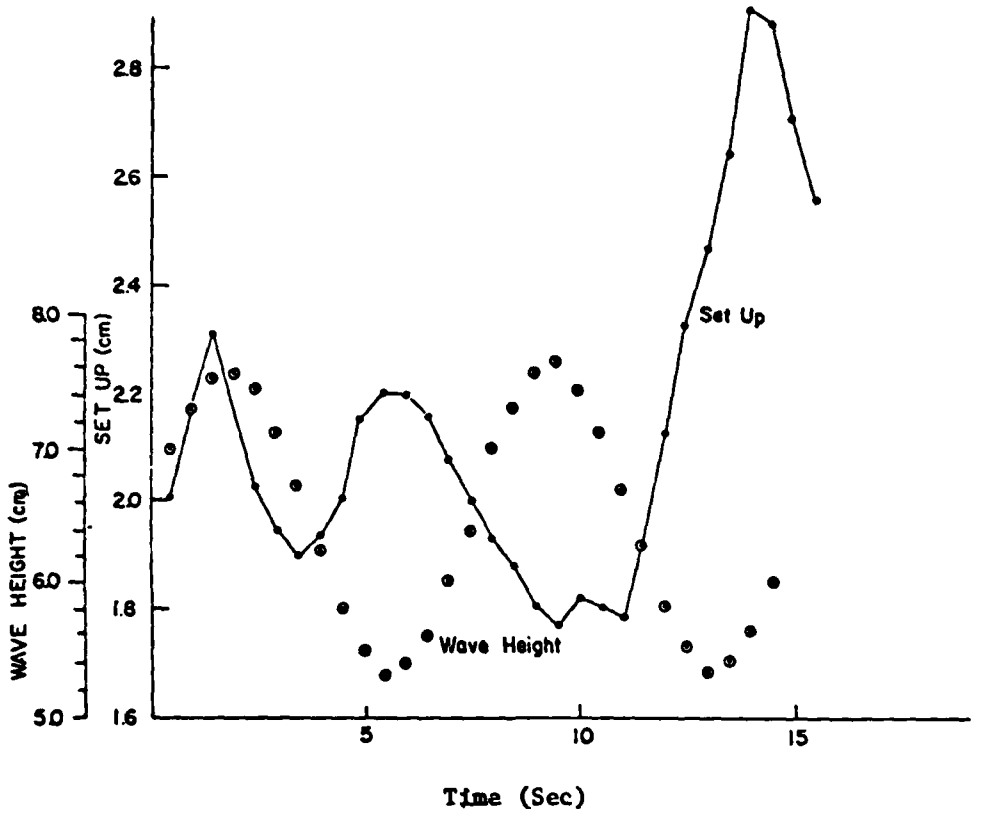


Figure 5-3. Resonance of Wave Channel Due to Forcing at Seiche Period

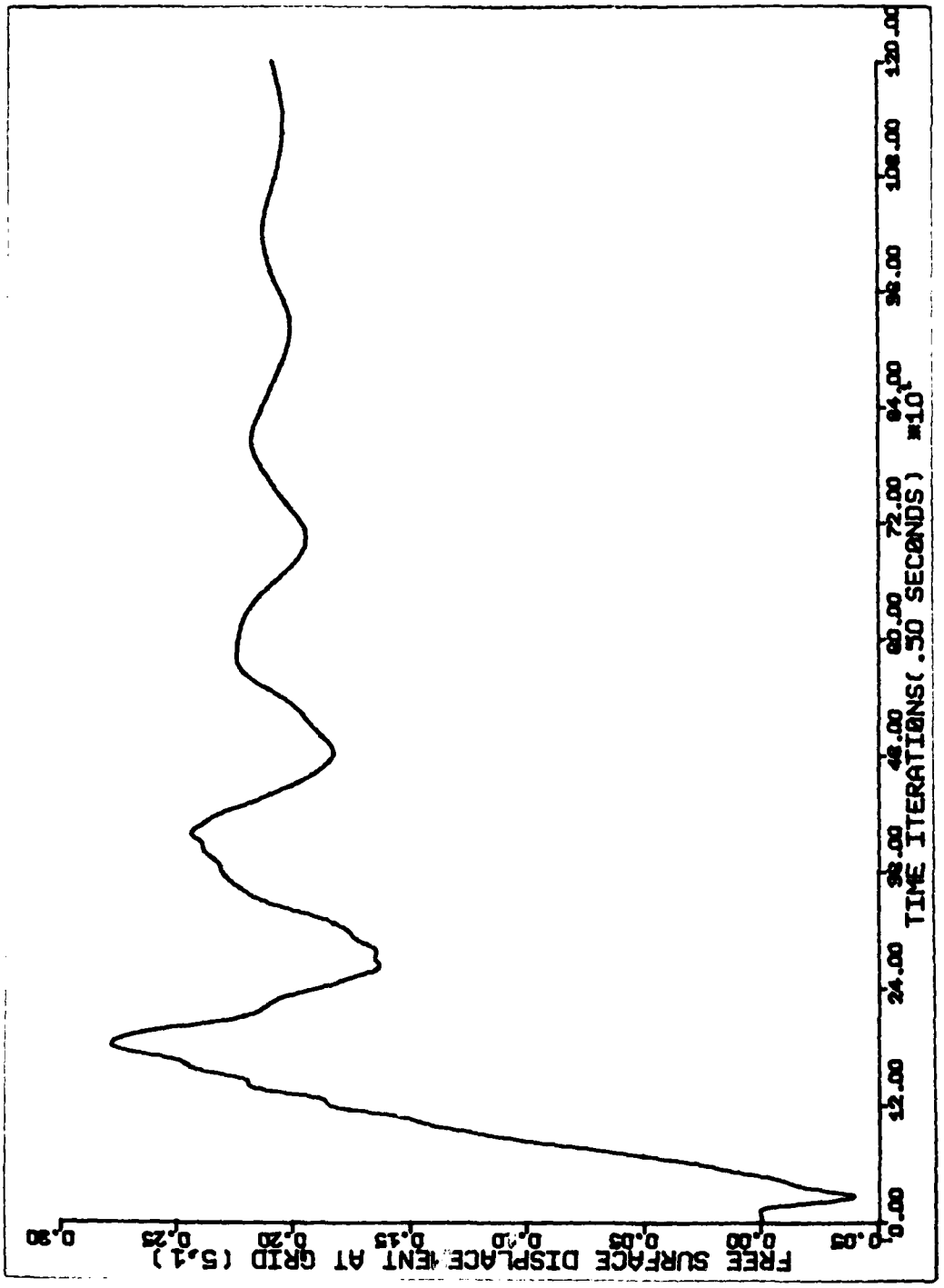


Figure 5-4. Inshore Mean Free Surface Displacement Versus Time for the Non-Linear Model Application to a Plane Beach

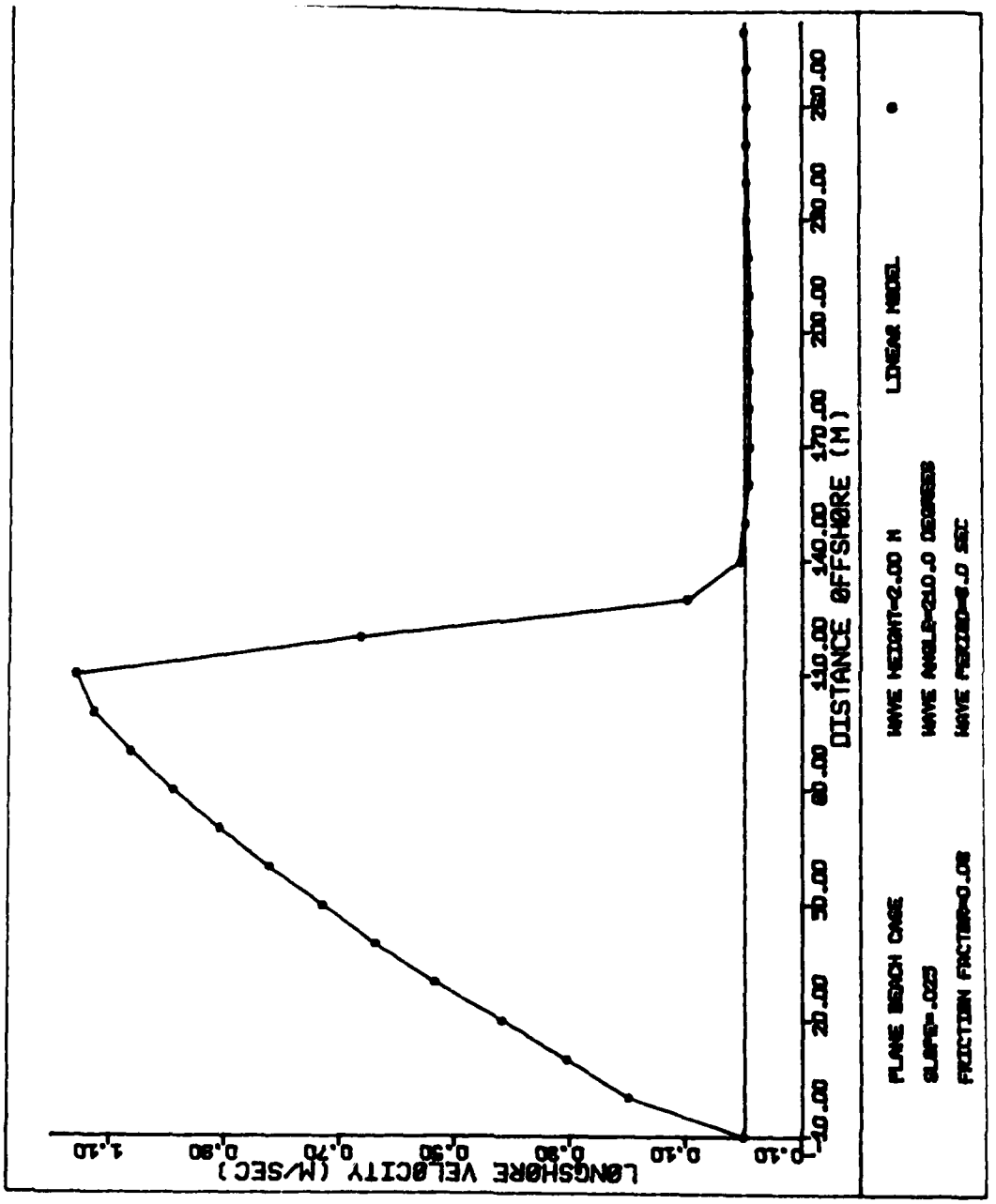


Figure 5-5. Longshore Current Profile for Plane Beach Linear Model

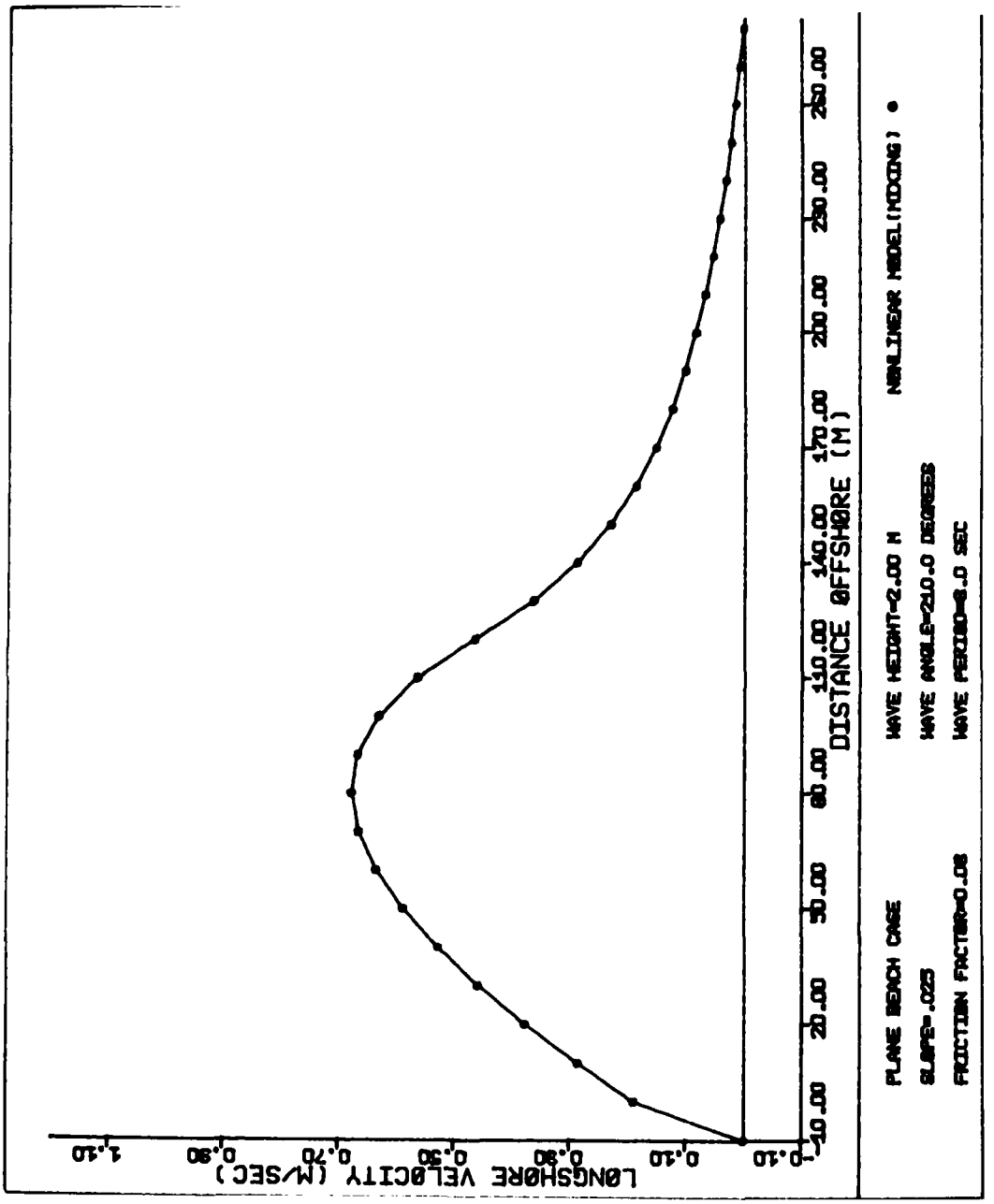


Figure 5-6. Longshore Current Profile for Plane Beach including Lateral Mixing
Nonlinear Model

The applications discussed here indicate that the model accurately reproduces the available analytic solutions for simplified physical situations.

2.2 Barred Profile Applications

In nature, beaches are often fronted by continuous or fragmented longshore bars. In order to model this situation, the nonlinear model was run using a barred profile (Figure 5-7) for the same wave conditions used to generate the plane beach results described above. For this case, the model was run neglecting the effect of lateral mixing, and including the effect; results are shown in Figures 5-8 and 5-9 respectively. The solution without lateral mixing demonstrates the major discrepancy found between theoretical solution and field data for the special case. As the model wave passes the crest of the offshore bar, it stops breaking since it responds instantaneously to the local bottom. In the absence of lateral mixing, no currents are generated in the trough between shore and bar. This result is in disagreement with field observation as shown recently by Allender et al. (1981) as well as other investigators. The inclusion of lateral mixing effects partially alleviates this discrepancy.

Two factors may contribute to the observations that currents in natural surf zones tend to be strongest in the trough between bar and shore. First, fragmented bars tend naturally to induce two dimensional flow patterns characterized by rip currents flowing seaward at the gaps in the bar system. These currents are driven by longshore variations in the set-up resulting from interaction between the incident waves and the non-uniform topography.

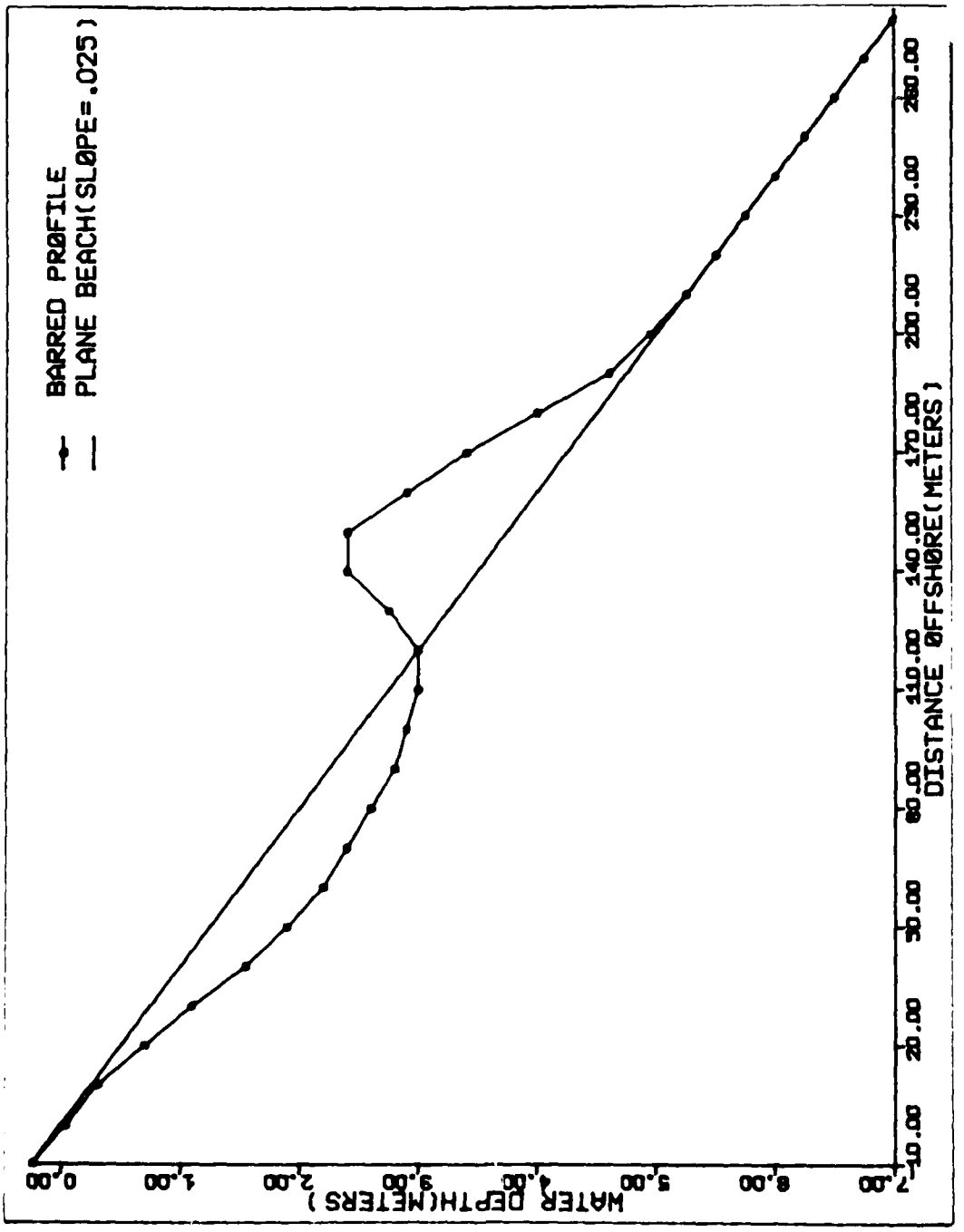
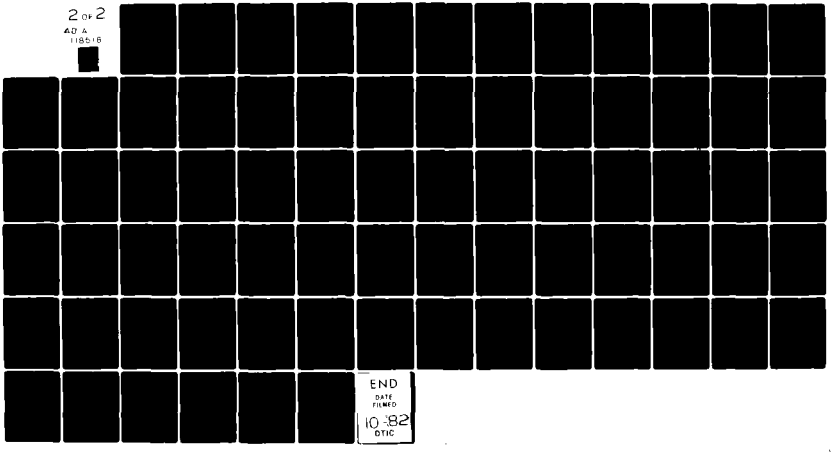


Figure 5-7. Barred Profile used for Results in Figures 5-8, 5-9.

AD-A118 518 DELAWARE UNIV NEWARK DEPT OF CIVIL ENGINEERING F/G 8/3
NUMERICAL MODELING OF THE NEARSHORE REGION.(U)
JUN 82 J T KIRBY, R A DALRYMPLE N00014-81-K-0297
UNCLASSIFIED CE-82-24 NL



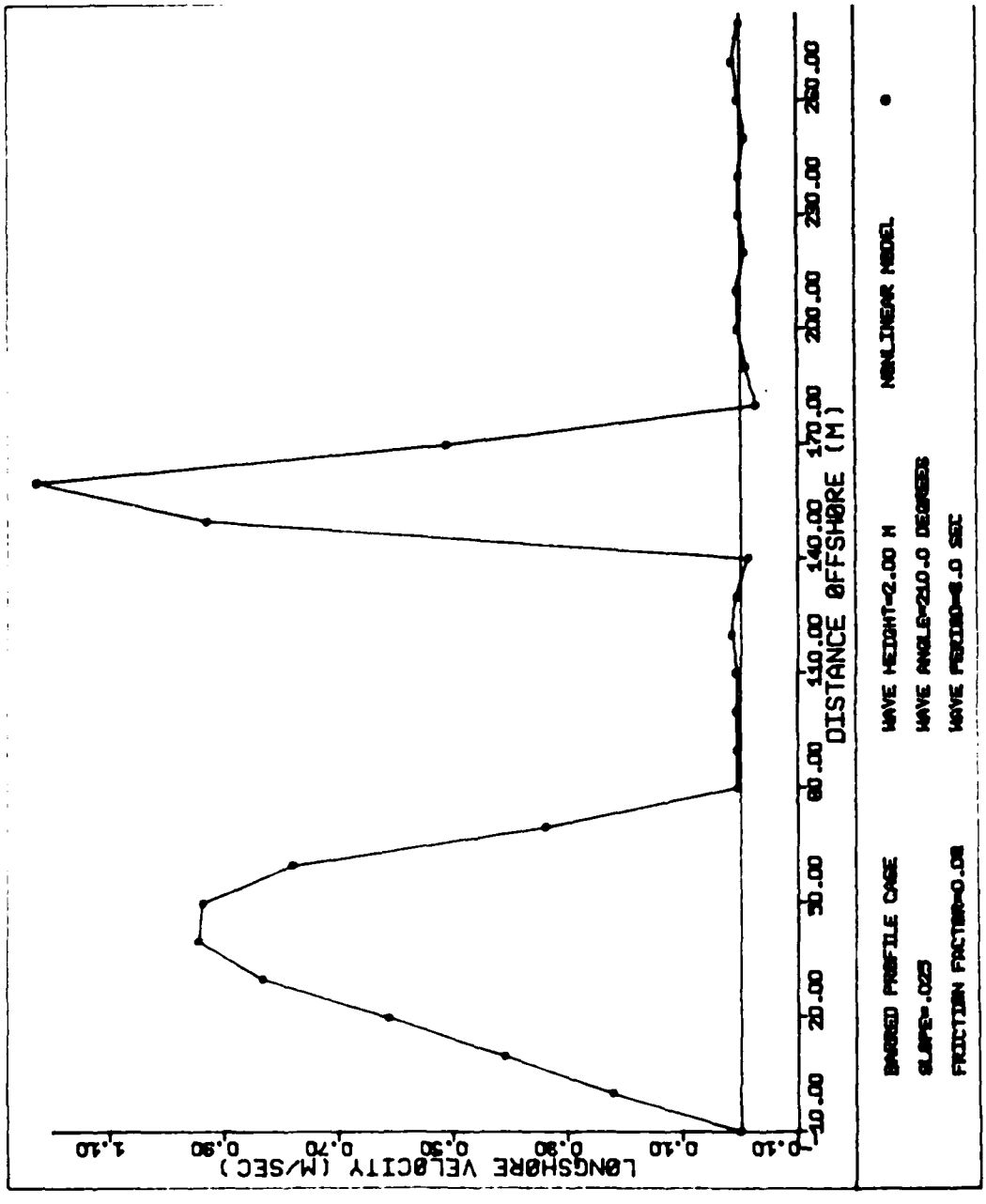


Figure 5-8. Longshore Current for Barred Profile
Nonlinear Model, No Lateral Mixing

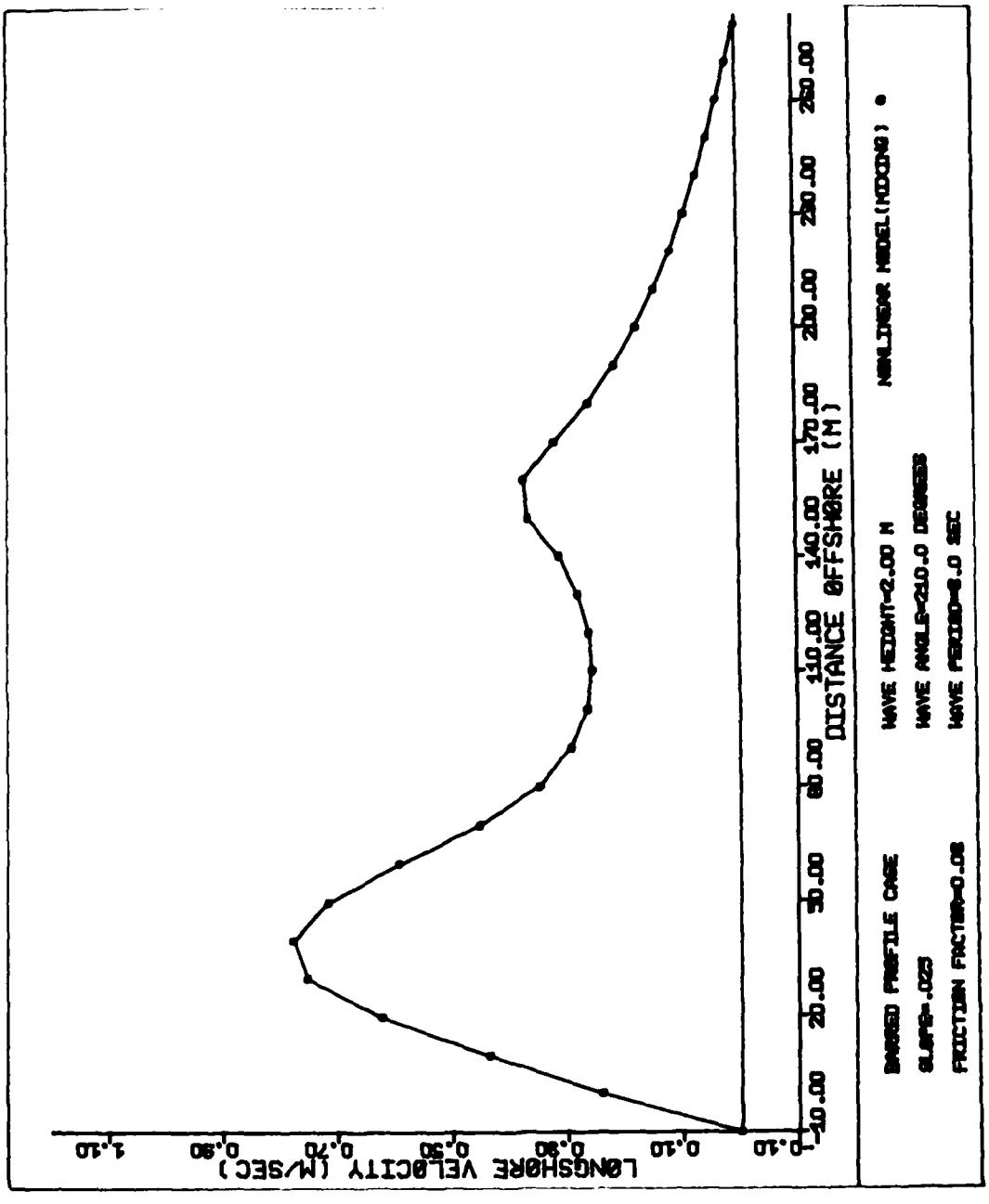


Figure 5-9. Longshore Current for Barred Profile
Nonlinear Model, Lateral Mixing Included

In these situations, currents flowing along the beach are driven by gradients in the mean water surface; the flow is naturally higher in the trough, where hydraulic resistance is lower. This mechanism for rip current maintenance has been discussed by Dalrymple (1978). It is possible that this mechanism will always bias the currents observed on real beaches.

The second possibility perhaps lies in the standard treatment of surf zone wave height as a function only of the local depth. The process of wave breaking initiates a turbulent flow pattern which is certain to have some time dependence, at least in its decay. A model for spilling breakers including this effect has been discussed by Longuet-Higgins and Turner (1974). It is possible, then, that an accurate picture of currents on a barred profile would require the inclusion of a continuation of wave energy decay in a time dependent manner past the point where the wave stops shoaling. Wave energy decay models have been used successfully by Divoky, LeMehaute and Lin (1970) in a study of wave height decay in the surf zone, and by Miller and Barcilon (1978) in a study of rip currents. However, present models do not allow for the cessation of breaking, which is required if the process of wave breaking and reformation is to be successfully treated.

2.3 Applications to the Laboratory Wave Basin

In order to model currents in a laboratory wave basin, the linear model described here was modified to include no flow boundary conditions at the longshore boundaries. (This option is not included in the present model.) Model tests were conducted to numerically approximate the experimental set-up and analytic theory of Dalrymple *et al.* (1977). In the physical experiment,

a plane beach was established at an angle of 15° to a flap-type wavemaker. Waves approached the beach over a flat bottom until they reached the front of the beach slope, where refraction effects began. Three cases were tested experimentally and theoretically; here, we restrict our attention to the case where the surf zone is bounded laterally by walls extending to infinity in the offshore direction. Experimentally determined streamlines are shown in Figure (5-10), in comparison to the analytic solution. In the numerical model, the physical situation was altered by extending impermeable walls in a direction normal to the beach; the waves, however, are allowed to propagate freely according to refraction governed by an infinite beach. The numerical model corresponded to the conditions used to develop the analytic solution. Velocity vectors calculated by the linear model are shown in Figure (5-11), in mirror image to the geometry in Figure (5-10). Longshore current velocities calculated by the linear model are compared to experimental values in Figure (5-12). The model is seen to underpredict currents in comparison to the analytic and experimental results.

2.4 Periodic Bottom Topography

In a study of rip currents caused by submerged on-offshore channels, Noda (1970) developed an equation for the depth which produces a channel oriented at an angle to the beach. The present models were tested using Noda's periodic bottom, given by

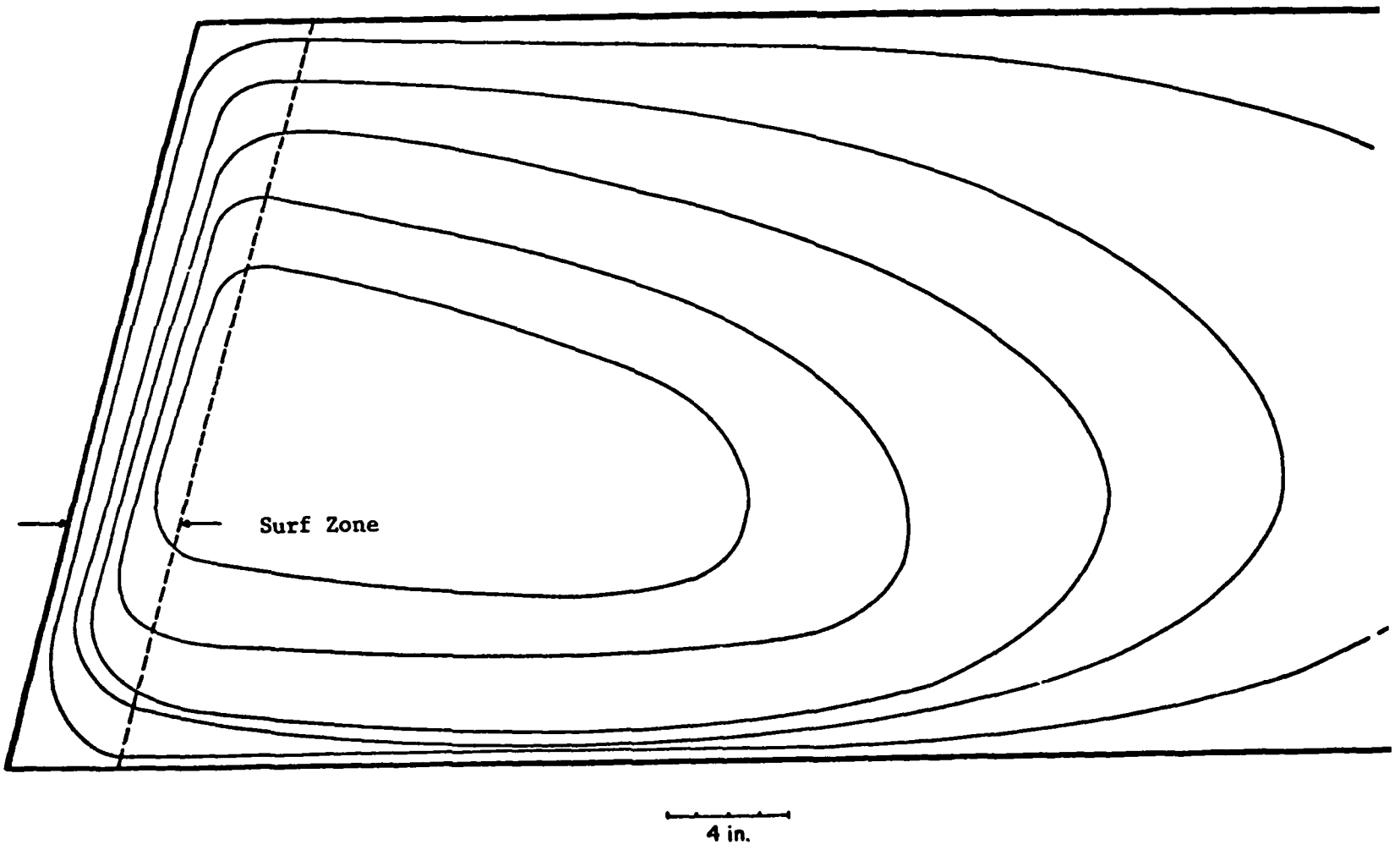


Figure 5-10. Experimental Strealines in the Wave Basin [From Dalrymple et al. (1977)].

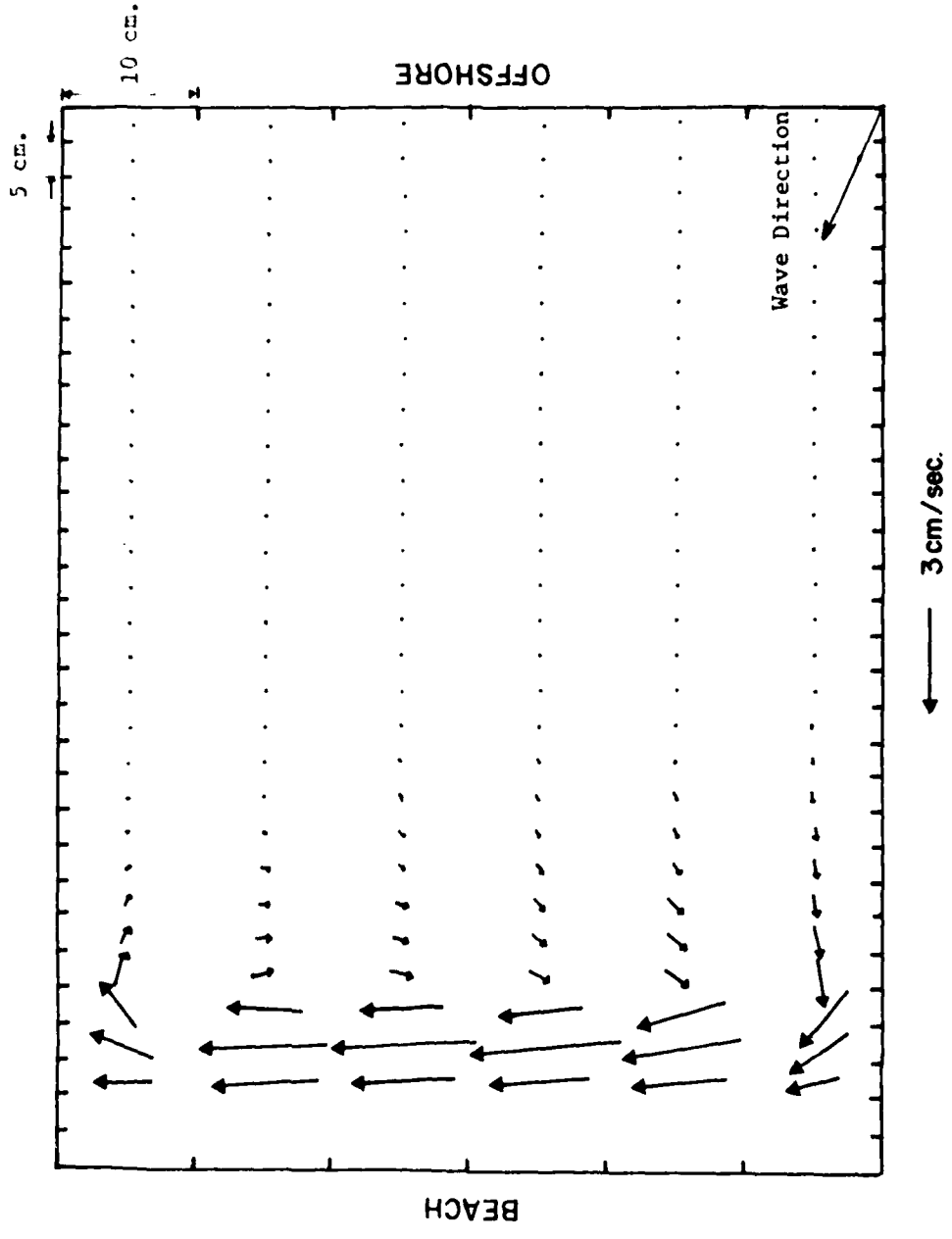


Figure 5-11. Calculated Velocity Vectors for Closed Basin Corresponding to Experimental Results of Figure 5-10

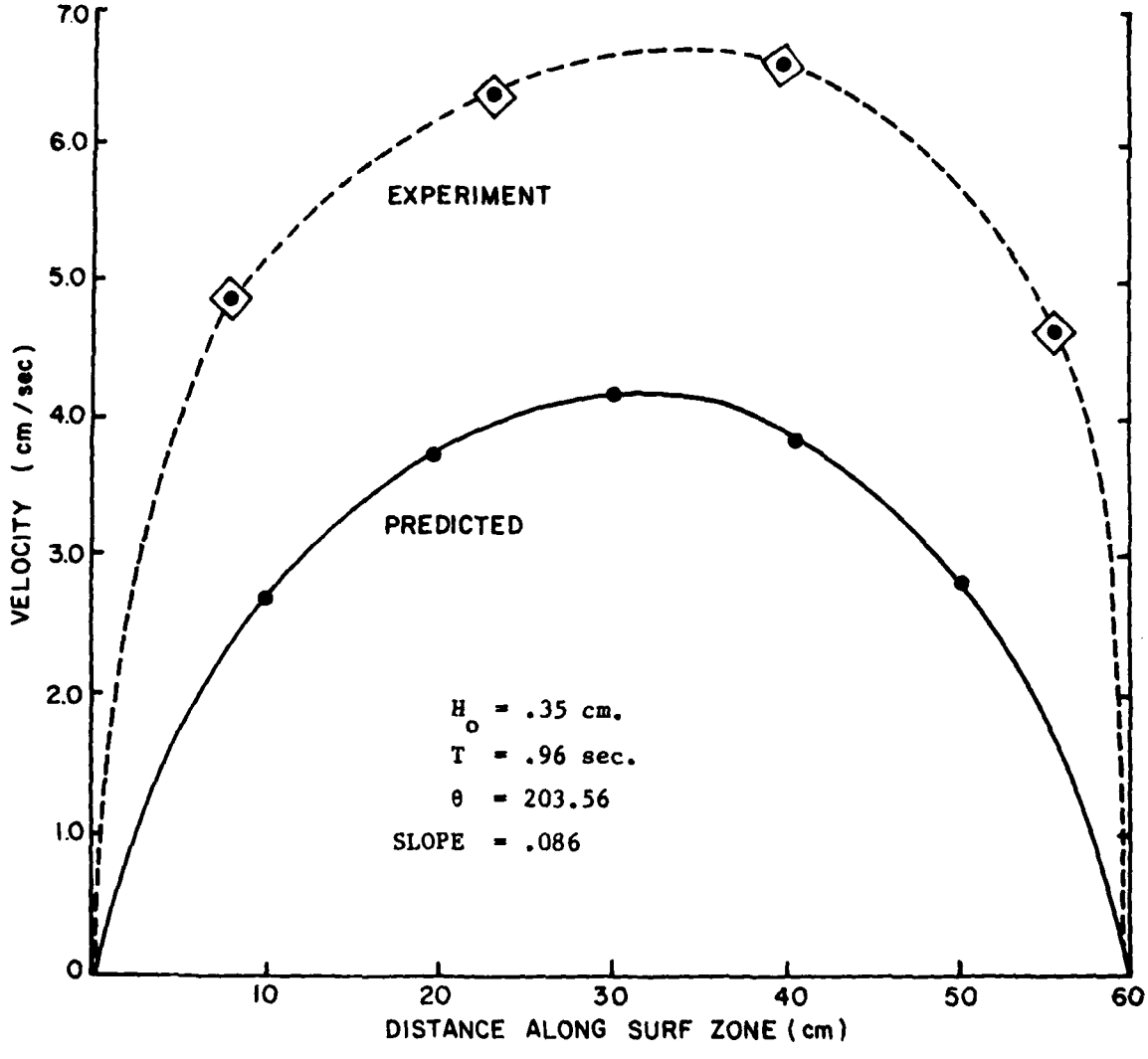


Figure 5-12. Predicted and Measured Longshore Velocities for the Closed Basin Test Case. Linear Model.

$$D_{i,j} = \begin{cases} -m(4 - i)\Delta x & i = 1,2,3,4 \\ m\Delta x\{1+A \exp[-3(\frac{x}{20})^{1/3}]\sin^{10}\frac{\pi}{\lambda}(y-x \tan \beta)\} & i > 4 \end{cases}$$

where x, y are coordinates of grid (i,j)

m = average beach slope = 0.025

λ = length of periodicity = 70 m

A = amplitude of bottom variation = 20

β = angle of rip channel to beach normal = 30°

The test contours are shown in Figure 5-13. The bottom topography was intended to model a field case studied by Sonu (1972). Current vectors calculated using the linear model are shown in Figure 5-14; current magnitude agree well with the field data reported in Noda *et al.* (1974), and are higher than the currents calculated in Noda *et al.*'s model, in which only 50% of the calculated current is used in determining the wave number due to numerical instability problems.

The Noda profile was also tested using the nonlinear model with lateral mixing (Figure 5-15). The effect of the extra terms modelled is to largely drop out the rip current clearly seen in the linear model results. Similar results have been obtained by Liu (1982), using a finite element approach, but the absence of a rip current under the given conditions is clearly at odds with the field data of Sonu (1972).

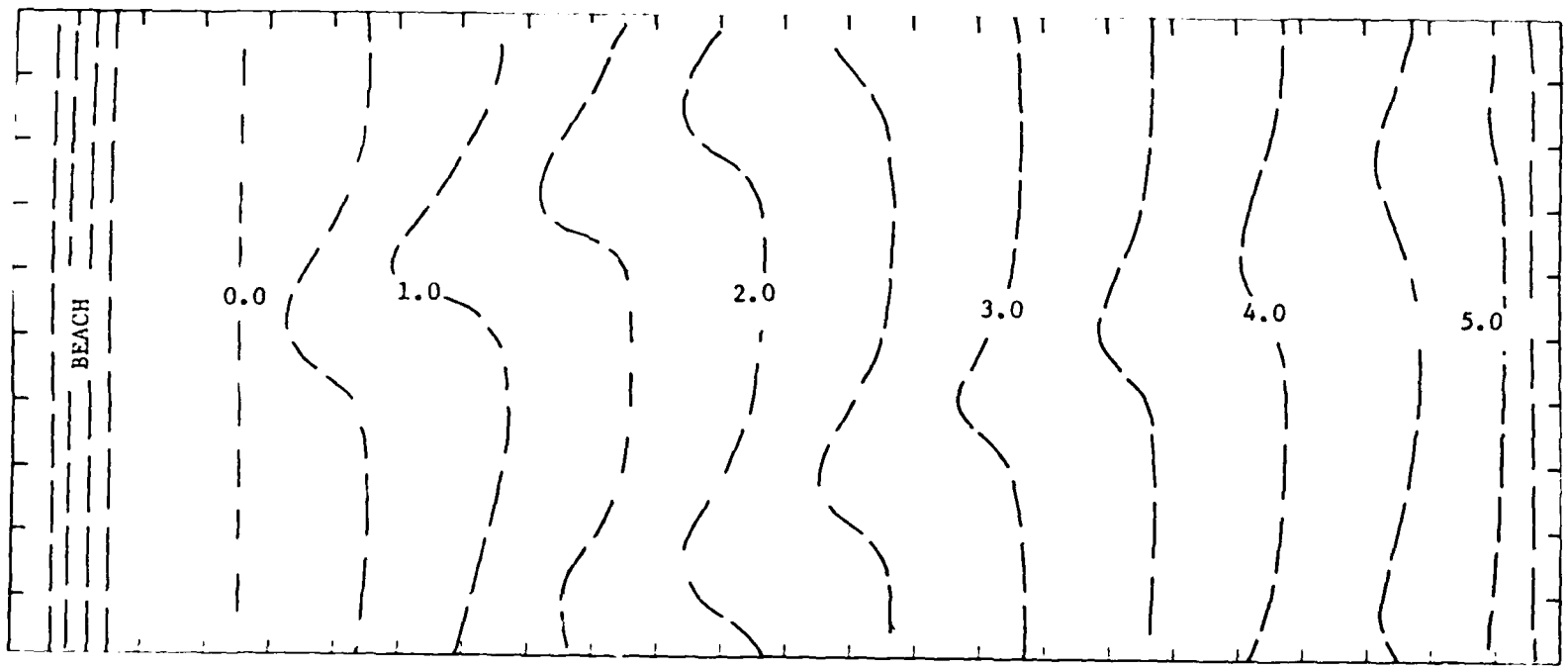


Figure 5-13. Depth Contours for the Periodic Bottom Due to Noda (1973).

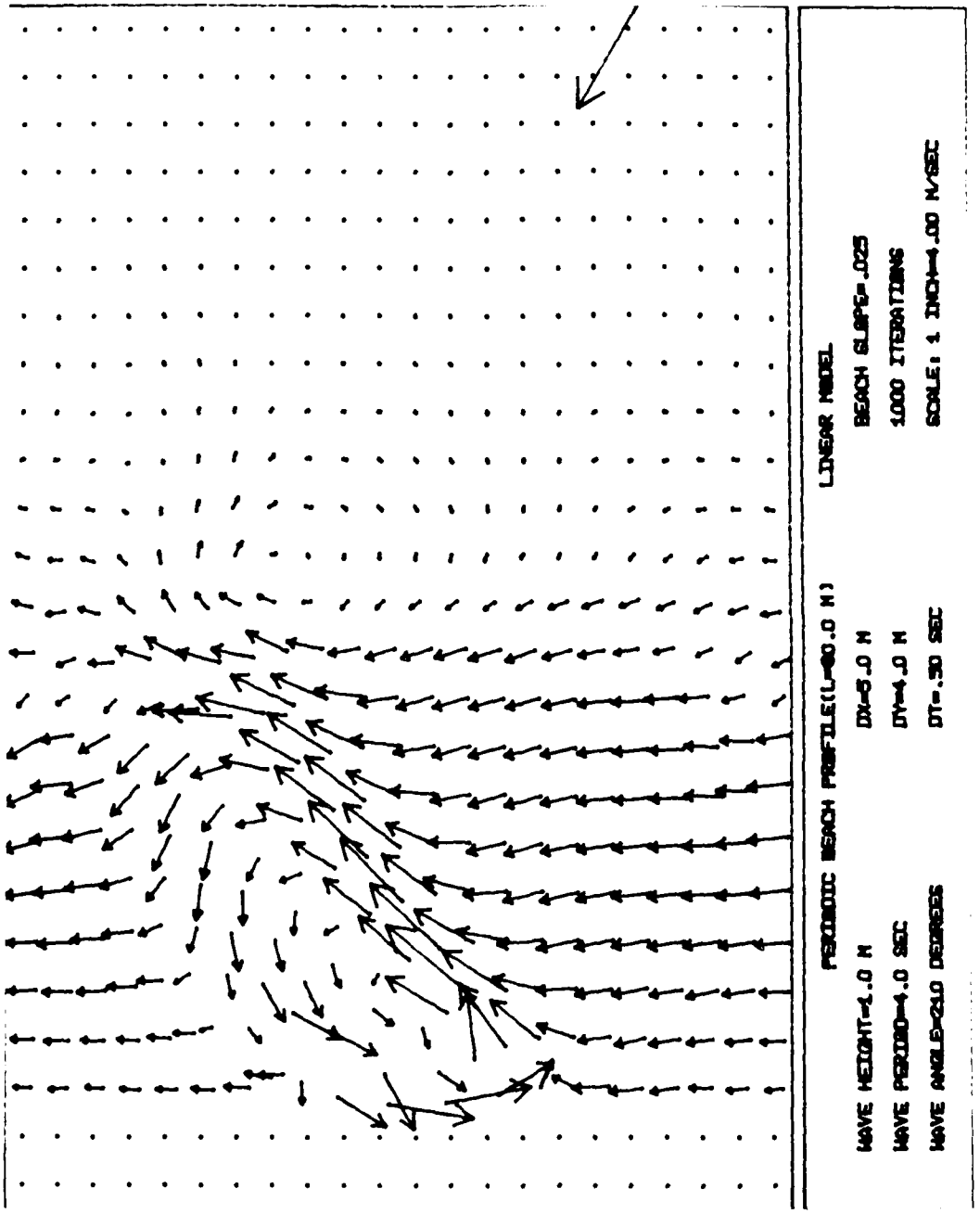


Figure 5-14. Current Vector Plot for Bottom Topography of Noda. Linear Model, No Lateral Mixing.

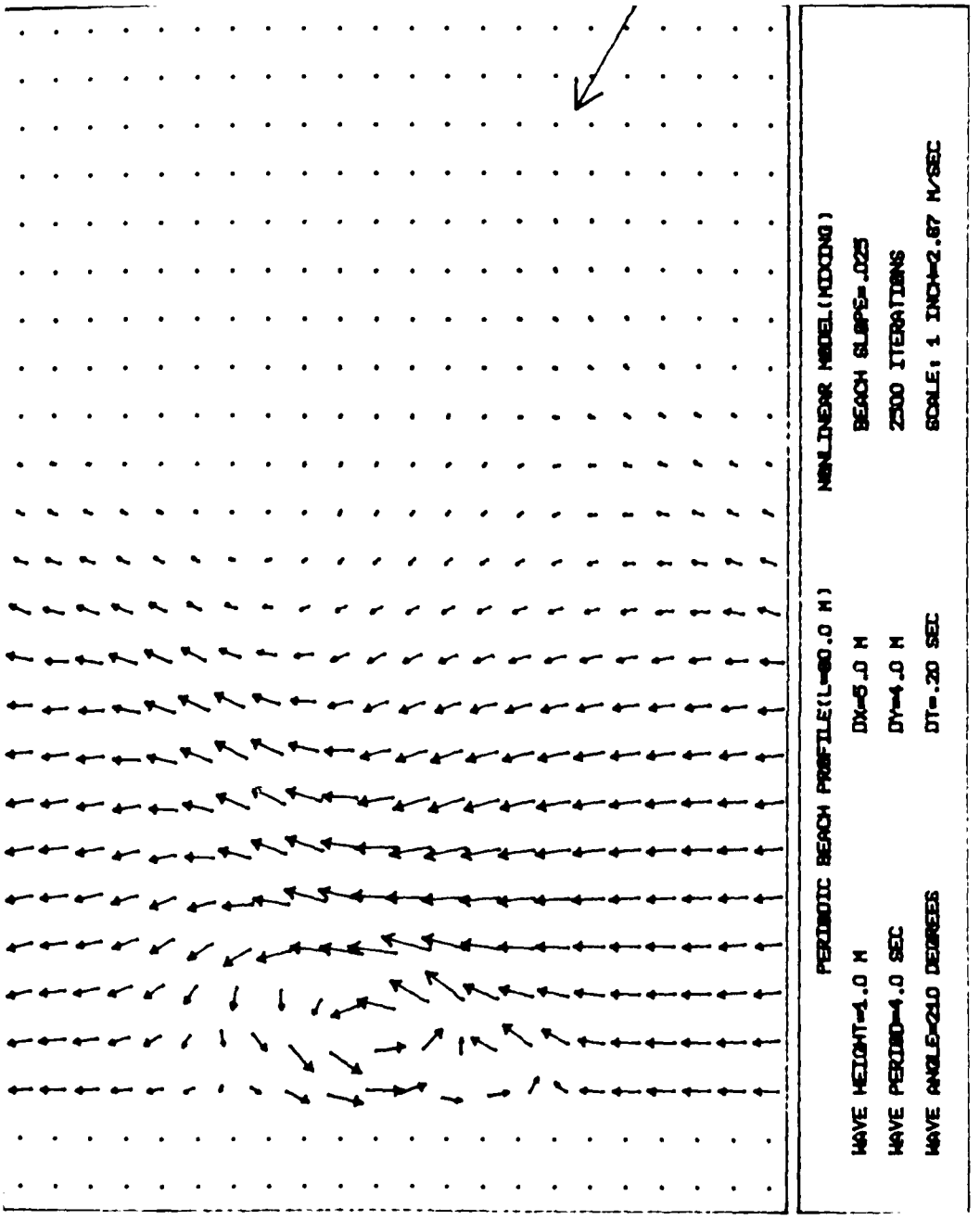


Figure 5-15. Current Vector Plot for Bottom Topography of Noda. Nonlinear Model, Lateral Mixing Included.

A second periodic topography has been investigated for the present study. It is well established that the presence of a nearly shore-normal channel deeper than the surrounding beach slope will tend to induce a rip current, with flow directed offshore in the channel. For the present case, a longshore-periodic perturbation consisting of a localized bulge, or relatively shallow region, has been added to an otherwise planar beach profile. The bottom is given by the relation

$$D_{i,j} = \begin{cases} -m\,(2 - i)\,\Delta x & i = 1 \\ m(i\Delta x)\left(1 - 0.9\cos^{6}\left\{\frac{j}{(N-1)}\,\pi\right\}\right)\exp(-0.015\,i\Delta x) & i \neq 1 \end{cases}$$

where

m = average beach slope = 0.02

i,j = grid coordinates

$(N-1)\Delta y$ = longshore periodicity

Results using both models are shown in Figures 5-16 and 5-17. The effect of the inclusion of convective acceleration terms in the nonlinear model is not apparent in the results, which may be a result of the counter-balancing effect of lateral mixing. The lateral mixing effect can be seen in the shoremost grid row. The overall effect modelled here can be explained most simply in terms of a longshore imbalance in the steady state set-up. As the bulge concentrates wave energy by refraction, a region of relatively high breaking waves is created. As water is pumped on shore by the gradient of the onshore radiation stress, a localized region of high set-up is created; this region in turn pumps the onshore-flowing water alongshore into the regions of low set-up. The waves over the region of the bulge then must continuously pump water onshore in an attempt to fill the surf zone up to the

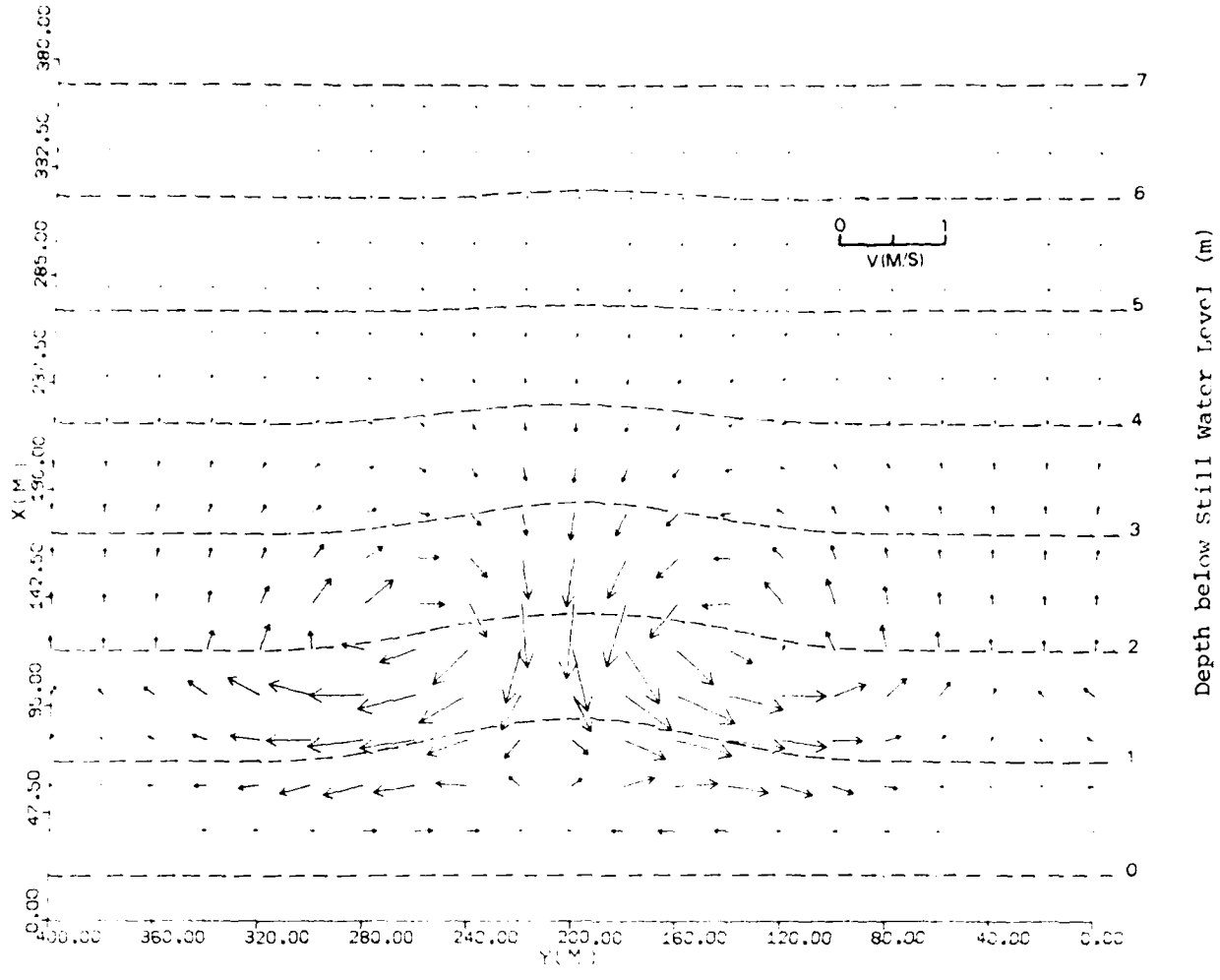


Figure 5-16. Currents Induced by Bulge on a Plane Beach Linear Model: $T = 10.0$ s, $H_o = 1.0$ m, $\theta_o = 180^o$.

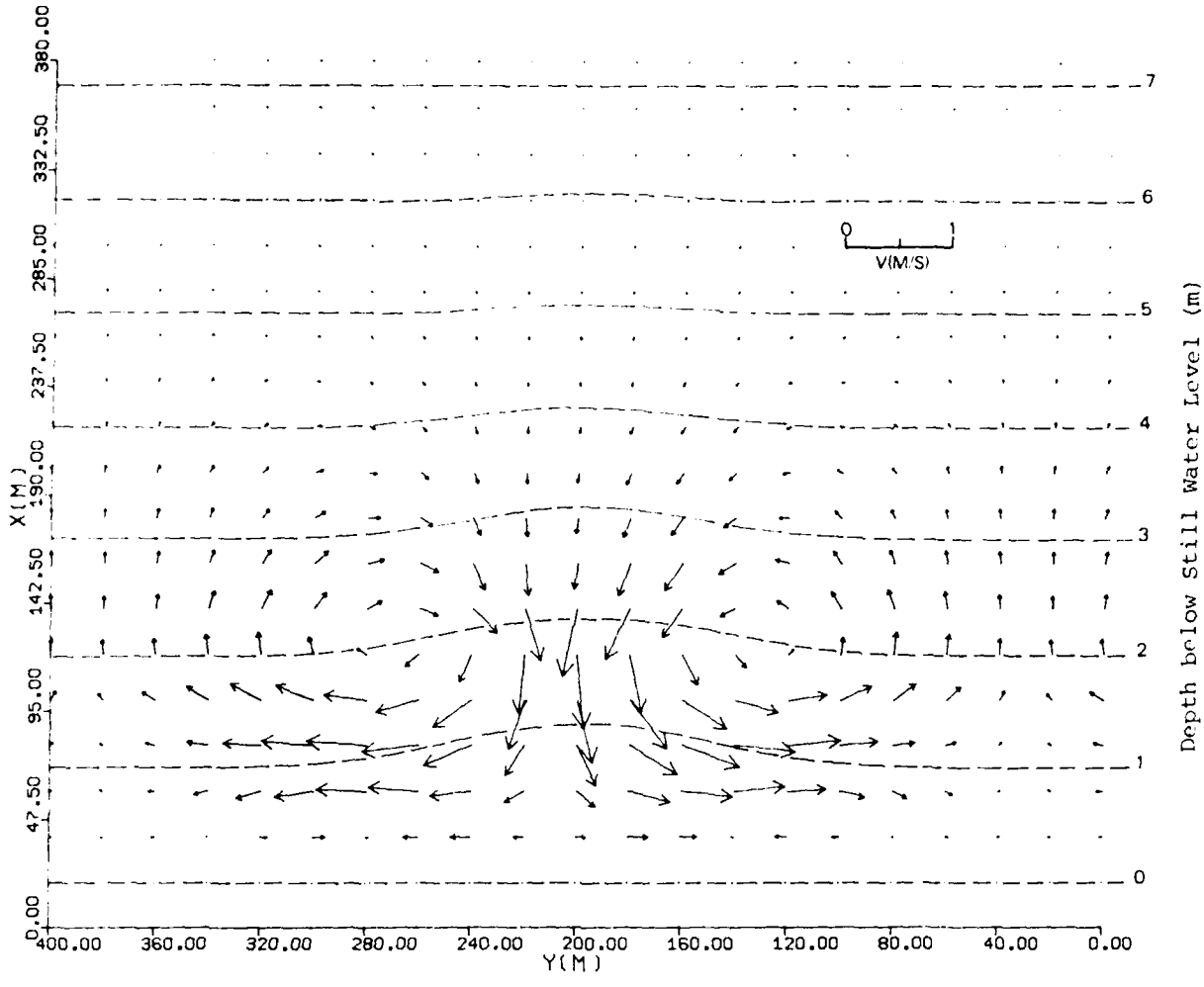


Figure 5-17. Currents Induced by Bulge on a Plane Beach Nonlinear Model: $T = 10.0$ s, $H_o = 1.0$ m, $\theta_o = 180^{\circ}$.

level determined by the decay of wave energy, leading to a steady state circulation pattern.

2.5 Intersecting Waves Application

Ebersole and Dalrymple have discussed the application of the model to the case of intersecting wave trains of the same frequency on a plane beach, which provides a mechanism for generating rip currents, as shown by Dalrymple (1975). The purpose of this application was to investigate the effect of the convective acceleration terms in the model. The following derivation closely follows the work of Dalrymple.

Given two intersecting wave trains A and B with amplitudes a and b and a common frequency, σ, in terms of the coordinate system shown in Figure 5-18, the free surface displacements for the two wave trains can be written as,

$$\eta_A = a \cos(k \cos \alpha x + k \sin \alpha y + \sigma t)$$

$$\eta_B = b \cos(k \cos \beta x + k \sin \beta y + \sigma t) \quad .$$

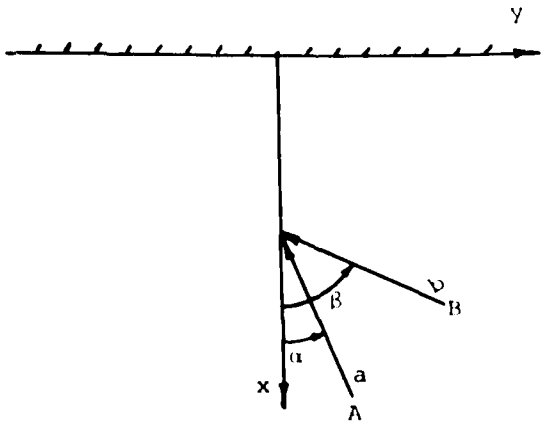


Figure 5-18. Wave Angles for Intersecting Waves.

The total free surface $\eta_T = \eta_1 + \eta_2$ can then be written as,

$$\eta_T = 2a \cos\{\frac{k}{2}(\cos\alpha + \cos\beta)x + \frac{k}{2}(\sin\alpha + \sin\beta)y + \sigma t\} \cdot$$

$$\cos\{\frac{k}{2}(\cos\alpha - \cos\beta)x + \frac{k}{2}(\sin\alpha - \sin\beta)y\}$$

$$+ (b-a)\cos\{k\cos\beta x + k\sin\beta y + \sigma t\} \quad . \qquad (5.2)$$

Using the linearized dynamic free surface boundary condition the velocity potential ϕ_T can be shown to equal,

$$\phi_T = \frac{2ag}{\sigma}\frac{\cosh k(h+z)}{\cosh kh}\sin\{\frac{k}{2}(\cos\alpha + \cos\beta)x + \frac{k}{2}(\sin\alpha + \sin\beta) + \sigma t\} \cdot$$

$$\cos\{\frac{k}{2}(\cos\alpha - \cos\beta)x + \frac{k}{2}(\sin\alpha - \sin\beta)y\}$$

$$+ \frac{(b-a)}{\sigma}\frac{\cosh k(h+z)}{\cosh kh}\sin\{k\cos\beta x + k\sin\beta y + \sigma t\} \quad .$$

From the velocity potential the total orbital velocities can be found from,

$$u = -\frac{\partial\phi}{\partial x} \quad , \quad v = -\frac{\partial\phi}{\partial y} \quad , \quad w = -\frac{\partial\phi}{\partial z} \quad .$$

The radiation stresses, which are essentially the forcing terms, are defined as,

$$S_{xx} = \int_{-h}^{0} \overline{\rho u^2}\,dz + \int_{-h}^{\eta} \overline{P}\,dz - \frac{1}{2}\rho g(h+\bar{\eta})^2 + \frac{1}{2}\rho g\,\overline{\eta^2}$$

$$S_{yy} = \int_{-h}^{0} \overline{\rho v^2}\,dz + \int_{-h}^{\eta} \overline{P}\,dz - \frac{1}{2}\rho g(h+\bar{\eta})^2 + \frac{1}{2}\rho g\,\overline{\eta^2}$$

$$S_{xy} = \int_{-h}^{0} \overline{\rho uv}\,dz$$

where

$$\overline{P} = \rho g(\bar{\eta}-z) + \frac{\partial}{\partial x}\int_{z}^{0} \overline{\rho uw}\,dz + \frac{\partial}{\partial y}\int_{z}^{0} \overline{\rho vw}\,dz - \overline{\rho w^2} \quad .$$

The radiation stresses are found to be given by

$$S_{xx} = \frac{\rho g}{4\sinh 2kh}\left[a^2\cos^2\alpha + b^2\cos^2\beta + 2ab\cos\alpha\cos\beta\cos\{2\psi\}\right] \cdot$$

$$\{2kh+\sinh 2kh\} - \frac{\rho g\, ab}{8\sinh 2kh}(\cos\beta - \cos\alpha)^2\cos\{2\psi\} \cdot$$

$$\{2kh\cosh 2kh-\sinh 2kh\} - \frac{\rho g\, ab}{8\sinh 2kh}(\sin\alpha-\sin\beta)^2\cos\{2\psi\} \cdot$$

$$\{2kh\cosh 2kh-\sinh 2kh\} - \frac{\rho g}{4\sinh 2kh}\left[a^2+b^2+2ab\cos\{2\psi\}\right] \cdot$$

$$\{\sinh 2kh-2kh\} + \rho gabcos^2\psi + \frac{1}{4}\rho g(b-a)^2$$

$$S_{yy} = \frac{\rho g}{4\sinh 2kh}\left[a^2\sin^2\alpha+b^2\sin^2\beta+2ab\sin\alpha\sin\beta\cos\{2\psi\}\right] \cdot \{2kh+\sinh 2kh\}$$

$$- \frac{\rho g\, ab}{8\sinh 2kh}(\cos\beta-\cos\alpha)^2\cos\{2\psi\} \cdot \{2kh\cosh 2kh-\sinh 2kh\}$$

$$- \frac{\rho g\, ab}{8\sinh 2kh}(\sin\alpha-\sin\beta)^2\cos\{2\psi\} \cdot \{2kh\cosh 2kh-\sinh 2kh\}$$

$$- \frac{\rho g}{4\sinh 2kh}\left[a^2+b^2+2ab\cos\{2\psi\}\right] \cdot \{\sinh 2kh-2kh\}$$

$$+ \rho gabcos^2\psi + \frac{1}{4}\rho g(b-a)^2$$

$$S_{xy} = \frac{\rho g}{4\sinh 2kh}\left[a^2\sin\alpha\cos\alpha+b^2\cos\beta\sin\beta+ab\cos\{2\psi\}\sin(\alpha+\beta)\right] \cdot$$

$$\{2kh+\sinh 2kh\}$$

where the expression "ψ" is defined as,

$$\psi \equiv \frac{k}{2}(\cos\alpha-\cos\beta)x + \frac{k}{2}(\sin\alpha-\sin\beta)y \quad .$$

The time independent mean free surface displacement, $\bar{\eta}$, is defined by

$$\bar{\eta} \equiv -\frac{1}{2g}\,\overline{\{u^2+v^2+w^2\}}_{z=0}$$

where "——" denotes the time average over one wave period. Substituting the expressions for the velocity components u, v, and w from the velocity potential ϕ_T, $\bar{\eta}$ can be written as,

$$\bar{\eta} = \frac{-k}{2\sinh 2kh}\,[a^2+b^2+2ab\cos\{2\psi\}\cdot(\cos(\alpha-\beta)\cosh^2 kh-\sinh^2 kh] \qquad (5.3)$$

where "ψ" is the same quantity defined previously. Notice that the mean free surface displacement is modulated in the x and y directions by,

$$\cos\{k(\cos\alpha-\cos\beta)x + k(\sin\alpha-\sin\beta)y\} \quad .$$

Using Snell's Law which states

$$k_o \sin\alpha_o = k \sin\alpha \quad \text{and} \quad k_o \sin\beta_o = k \sin\beta \quad ,$$

and using the fact that $k_o \equiv \frac{2\pi}{L_o}$, where "o" denotes deep water values for the wave length, L, and the wave angles, α and β, we see that there is a periodicity of the mean displacement in the longshore direction with a periodic spacing, ℓ, given by,

$$\ell = \frac{L_o}{\sin\alpha_o - \sin\beta_o}$$

This periodicity in water level and wave height causes water to be driven from regions of high mean water level displacement to regions of lower displacement, resulting in the formation of circulation cells.

In order to attempt to model this phenomena, certain simplifications to the model had to be made. Since the refraction and shoaling routines borrowed from the work of Noda, et al., could not treat more than one wave train, they were replaced with routines governed by Snell's Law neglecting wave-current interaction. Again a quadratic, "exact," bottom friction was used including velocities due to mean currents and, this time, the two wave trains. In the momentum equations the advective acceleration terms were retained, horizontal mixing was neglected, and the radiation stresses were calculated using the results presented earlier in this section. The wave height used in calculating the radiation stresses and the bottom frictional stresses is given by

$$H_T = 2.0 \sqrt{a^2+b^2+2ab \cos[k(\cos\alpha-\cos\beta)x + k(\sin\alpha-\sin\beta)y]} \ .$$

Two runs are presented here using different combinations of wave heights and wave angles. The remainder of the input data for both runs, however, was the same. The waves were run on a plane beach with a slope of 0.025. The planform area of interest was comprised of 25 grids in the x direction with an Δx grid size of 5.0 meters, and 21 grids in the longshore direction with a Δy grid size of 4.0 meters. The time step was chosen to be 0.2 seconds and the model was run for 1500 iterations for both cases. A wave period of 7.159366 seconds was used, which resulted in theoretical rip spacings of 80.0 meters. The bottom friction factor was set equal to 0.12 to allow the system to reach steady state after the 1500 iterations and to decrease the magnitude of the resultant currents.

The first case used waves of equal heights and equal angles to either side of the beach normal. The deep water wave heights were 0.25 meters and

the deep water angles were ± 30.0 degrees. For this case, referring to Eq. (5.3), $a=b$ and $\alpha=-\beta$ resulting in a free surface displacement given by

$$\eta_T = 2a \cos(k_o \sin \alpha_o y)\cos(k \cos \alpha x + \sigma t) \quad .$$

This free surface describes a wave train moving in the -x direction with a modulated wave height that is periodic in the longshore direction only. The periodicity in wave height is the driving mechanism producing the rip current perpendicular to the beach, as shown in Figure 5-19. Note the constricted width of the rip current in relation to the width of the inflow region. This is a result of the convective acceleration terms. Also note the weak rip head where the currents diverge from the rip axis and return towards shore.

In the second case, the waves, A and B, had different heights, 0.1 and 0.4 meters, and wave angles of 30.0 and -30.0 degrees, respectively. The resulting circulation pattern is shown in Figure 5-20, and consists of a meandering current with alternating regions of strong and weak longshore velocity along the beach. This circulation would lend itself well to the formation of rythmic beach features. Looking at Eq. (5.2), we see that there is a non-zero term,

$$(b-a)\cos\{k \cos \beta x + k \sin \beta y + \sigma t\}$$

which is a wave train at an angle to the beach normal with height 2(b-a). This wave is present in addition to the normal wave train with the modulated height from the first case, causing a longshore current which is superimposed on the cellular circulation.

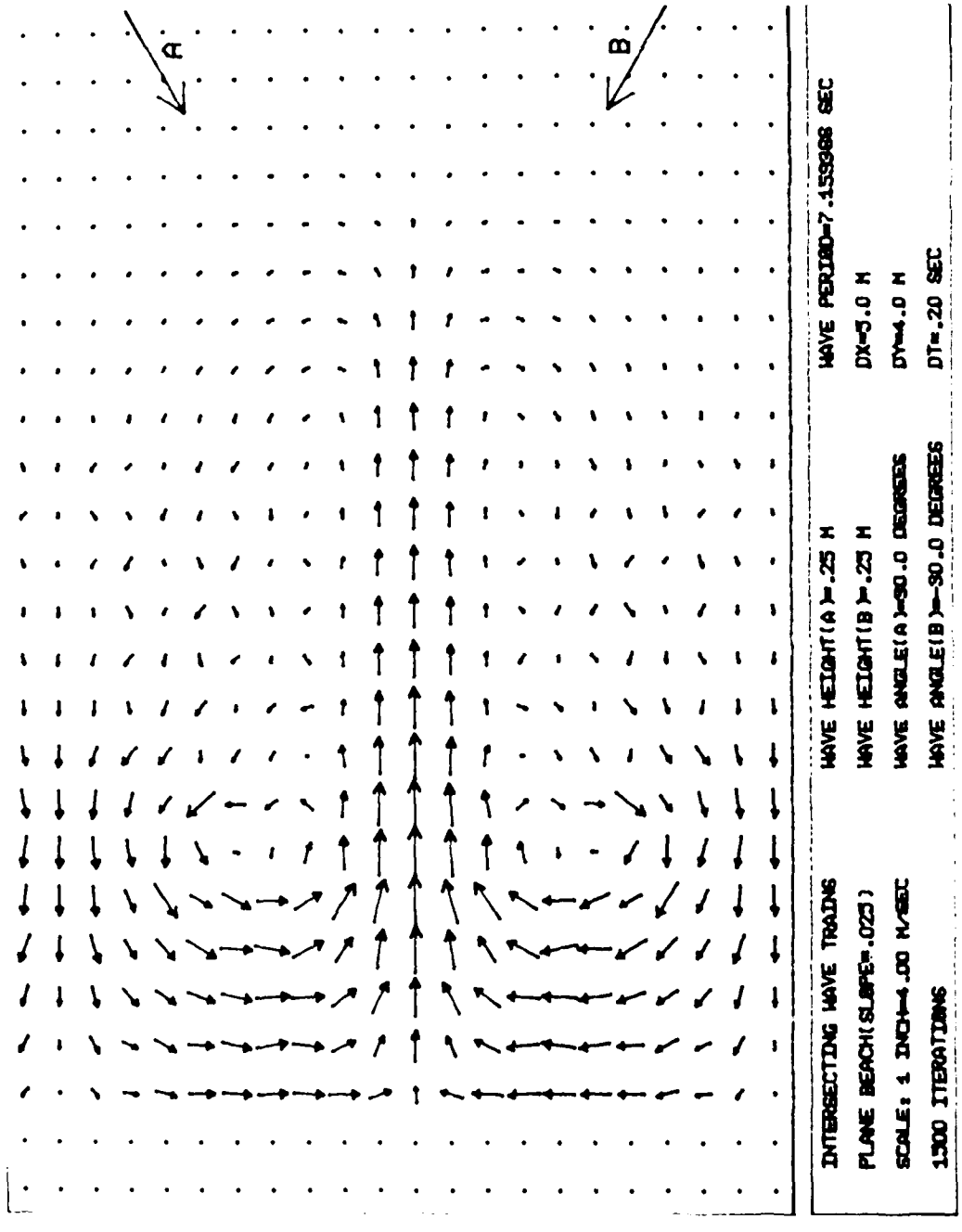


Figure 5-19. Current Vector Plot for Intersecting Waves, Case 1. Nonlinear Model

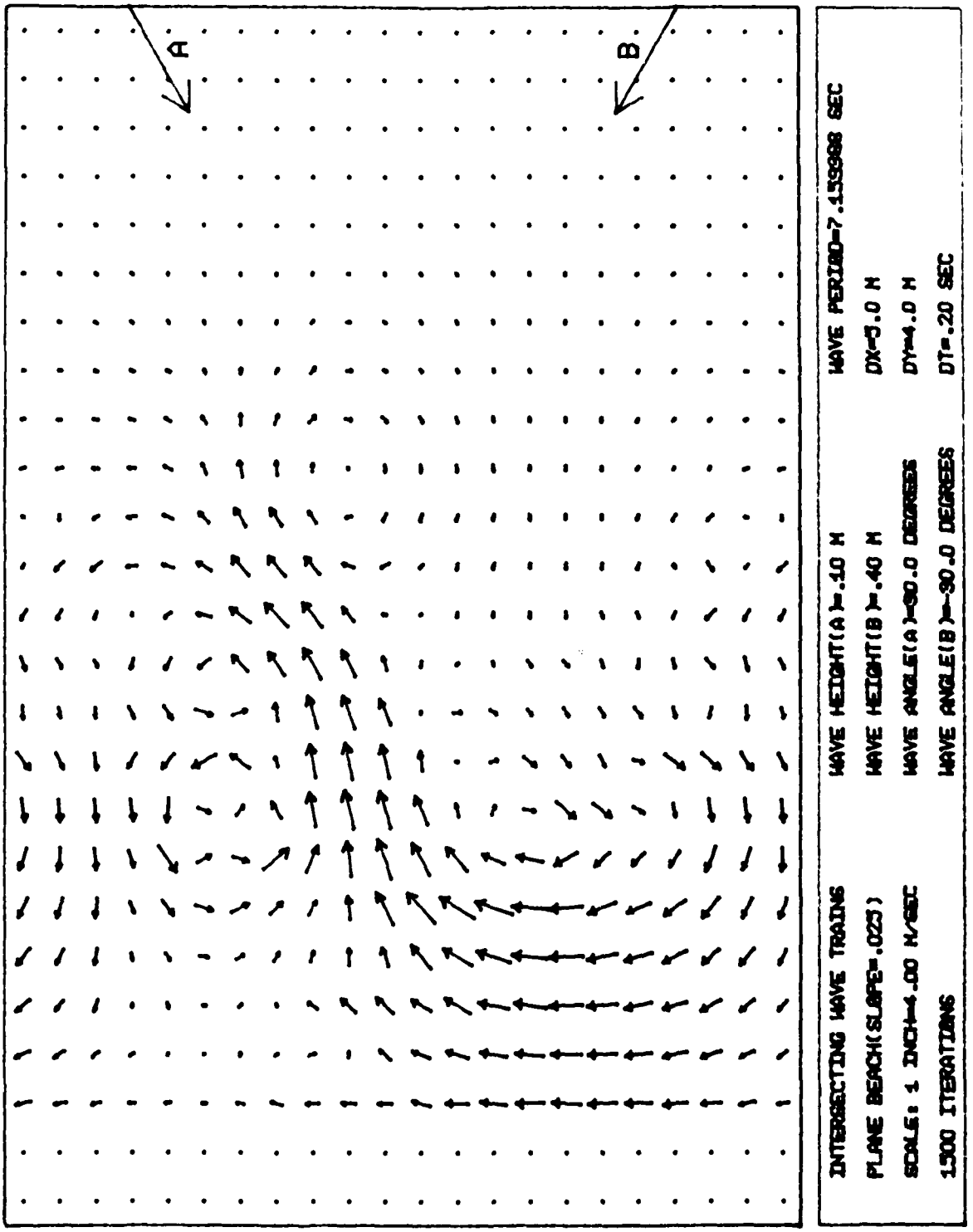


Figure 5-20. Current Vector Plot for the Meandering Circulation Pattern. Nonlinear Model.

Chapter VI

CONCLUSIONS

In this report, the theoretical background and numerical formulation of two models for nearshore circulation have been reviewed. A review of the application of the models to plane beach problems has shown that the models successfully reproduce the characteristics of known analytic solutions. The comments of Basco (1981), who expressed the opinion that the models are invalid due to the effect of numerical viscosity on the calculated velocity profiles, can be seen to be incorrect upon inspection of Figure 5-5, which demonstrates the steep drop in longshore current at the breaker line in the absence of a prescribed lateral mixing. In addition, Figure 5-1 shows that the model reproduces the sharp drop in wave set down at the breaker line, further indicating that the model is capable of reproducing the distinct features of analytic solutions, if the grid scheme chosen is fine enough to resolve the features. It is apparent that any numerical diffusion of solutions for $\bar{\eta}$ or V is confined to one or two grid rows, and can be made insignificant by choosing the grid size small enough. It should also be remarked that the slight decrease in the longshore velocity below the purely linear profile predicted by Longuet-Higgins (1970a) has been explained to be a result of wave-current interaction (Dalrymple, 1980) included in both models, and of finite angle of wave incidence (Liu and Dalrymple, 1978; Krauss and Sasaki,1979), included in the nonlinear model. In conclusion, there appears to be no discrepancy between numerical solutions using a fine grid and the corresponding analytic solutions.

In practical applications, grid spacings are often chosen which lead to some false numerical averaging of the solutions. In the linear model, this averaging manifests itself most obviously as a smoothing of the longshore profile across the breaker line in the same manner as would be induced by lateral mixing effects (see Figure 4-14). This indicates that the mixing coefficients included in the nonlinear model should be smaller than expected on physical grounds in order to compensate for the numerical effects. With the choice of a friction factor based on the formulae of Jonsson (1966) or Kajiura (1968), the model is then seen to produce realistic current profiles, both in form and magnitude, for the field site chosen. However, it should be noted that no detailed data set currently exists which satisfies the requirement of a monochromatic input wave condition, and which represents a field case for which model results are fairly reliable, such as the absence of a large longshore bar.

The numerical models are shown to be successful predictors of nearshore dynamics in situations dominated by refraction and weak wave-current interaction effects. In this regard, it is noted that the model as currently formulated cannot handle certain effects, such as the influence of finite barriers in the wave field, or strong interaction of waves and opposing currents. The modelling of both of these effects requires the inclusion of a capability for handling wave diffraction.

Finally, the present models require an input wave condition based on a monochromatic wave, whereas wave trains in nature tend to include a spectrum of waves. In the case where the incident wave spectrum is sufficiently narrow-banded in frequency, the incident wave can be represented as a wave of single frequency with a modulated amplitude; such an effect can

be modelled after modification of the programs to handle a time varying deepwater wave height. However, modelling of broad-banded spectra at the level of the treatment here remains a complex problem.

REFERENCES

Allender, J. H., J. D. Ditmars, A. A. Frigo, and R. A. Paddock (1981), "Evaluation of a Numerical Model for Wave- and Wind-Induced Nearshore Circulation Using Field Data," Report NUREG/CR-1210, ANL/EES-TM-154, Argonne National Laboratory, Argonne, Illinois.

Basco, D. R. (1981), "Surf Zone Currents," U. S. Army Corps of Engineers, Coastal Engin. Res. Center (in preparation).

Birkemeier, W. A., and R. A. Dalrymple (1976), "Numerical Models for the Prediction of Wave Set-up and Nearshore Circulation," ONR Tech. Rept. 1, Ocean Engineering Rept. 3, Dept. of Civil Engineering, University of Delaware, Newark, DE.

Blumberg, A. F. (1977), "Numerical Tidal Model of Chesepeake Bay," ASCE J. Hydraulics Division 103, HY1, 1-9.

Bowen, A. J., and D. L. Inman (1974), "Nearshore Mixing Due to Waves and Wave-Induced Currents," Rapp. P.-v. Réun. Cons. int. Explor. Mer. 167, 6-12.

Bowen, A. J., D. L. Inman, and V. P. Simmons (1968), "Wave Set-Down and Set-up," J. Geophys. Res. 73(8), 2569-2577.

Dalrymple, R. A. (1975), "A Mechanism for Rip Current Generation on an Open Coast," J. Geophys. Res. 80(24), 3485-3487.

Dalrymple, R. A. (1978), "Rip Currents and their Causes," Proc. 16th Int. Conf. on Coastal Engineering, Hamburg.

Dalrymple, R. A. (1980), "Longshore Currents with Wave Current Interation," ASCE J. Waterway, Port, Coast. and Ocean Div. 106 WW3, 414-420.

Dalrymple, R. A., R. A. Eubanks, and W. A. Birkemeier, (1977), "Wave-Induced Circulation in Shallow Basins," ASCE J. Waterway, Port, Coast. and Ocean Div. 103 WW1, 117-135.

Dalrymple, R. A., and P. L.-F. Liu (1978), "Waves over Soft Muds: A Two Layer Fluid Model," J. Phys. Oceanogr. 8(6), 1121-1131.

Dalrymple, R. A., and C. J. Lozano (1978), "Wave-Current Interaction Models for Rip Currents," J. Geophys. Res. 83(C12), 6063-6071.

Divokey, D., B. LeMehaute, and A. Liu (1970), "Breaking Waves on Gentle Slopes," J. Geophys. Res. 75(9), 1681-1692.

Ebersole, B. A., and R. A. Dalrymple (1979), "A Numerical Model for Nearshore Circulation including Convective Accelerations and Lateral Mixing," ONR Tech. Rept. 4, Ocean Engineering Rept. 21, Dept. of Civil Engineering, University of Delaware, Newark, DE.

Gable, C. G. (ed.) (1979), "Report on Data from the Nearshore Sediment Transport Study Experiment at Torrey Pines Beach, California, November-December 1978," IMR Ref. No. 79-8, Institute of Marine Resources, University of California, La Jolla, CA.

Hildebrand, F. B. (1976), Advanced Calculus for Applications, Prentice-Hall.

James, I. D. (1974), "Non-linear Waves in the Nearshore Region: Shoaling and Set-up," Est. Coast. Mar. Sci. 2, 207-234.

Jonsson, I. G. (1966), "Wave Boundary Layers and Friction Factors," Proc. 10th Int. Conf. on Coastal Engineering, Tokyo, 127-148.

Kajiura, K. (1968), "A Model of the Bottom Boundary Layer in Water Waves," Bull. Earthquake Res. Inst. 46, 75-123.

Kraus, N. C., and T. O. Sasaki (1979), "Influence of Wave Angle and Lateral Mixing on the Longshore Current," Marine Sci. Comm. 5, 99-126.

Kurihara, Y. (1965), "On the Use of Implicit and Iterative Methods for Time Integration of the Wave Equation," Monthly Weather Rev. 93(1), 33-46.

LeBlond, P. H., and C. L. Tang (1974), "On Energy Coupling between Waves and Rip Currents," J. Geophys. Res. 79(6), 811-816.

Lilly, D. K. (1965), "On the Computational Stability of Numerical Solutions of Time-Dependent Non-Linear Geophysical Fluid Problems," Monthly Weather Rev. 93(1), 11-26.

Liu, P. L.-F. (1973), "Damping of Waves over a Porous Bed," ASCE J. Hydraulics Div. 99, HY , 2263-2271.

Liu, P. L.-F. (1982), "Finite Element Modelling of Breaking Wave-Induced Nearshore Currents," Proc. Nearshore Current Model Workshop, CERC, (in preparation) Newark, Del.

Liu, P. L.-F., and R. A. Dalrymple (1978), "Bottom Frictional Stresses and Longshore Currents Due to Waves with Large Angles of Incidence," J. Mar. Res. 36(2), 357-375.

Longuet-Higgins, M.S. (1970a), "Longshore Currents Generated by Obliquely Incident Sea Waves, 1," J. Geophys. Res. 75(33), 6778-6789.

Longuet-Higgins, M.S. (1970b), "Longshore Currents Generated by Obliquely Incident Sea Waves, 2," J. Geophys. Res. 75(33), 6790-6801.

Longuet-Higgins, M. S., and Stewart, R. W. (1969), "Radiation Stress in Water Waves: A Physical Discussion with Applications," Deep Sea. Res. 4, 529-563.

Longuet-Higgins, M. S., and J. S. Turner (1974), "An 'Entraining Plume' Model of a Spilling Breaker," J. Fluid Mech. 63(1), 1-20.

McDougall, W. G. (1979), "A Numerical Model to Predict Surf Zone Dynamics," M.C.E. Thesis, Dept. of Civil Engineering, University of Delaware, Newark, DE.

Miller, C., and A. Barcilon (1978), "Hydrodynamic Instability in the Surf Zone as a Mechanism for the Formation of Horizontal Gyres," J. Geophys. Res. 83, 4107-4116.

Noda, E. K. (1973), "Rip Currents," Proc. 13th Int. Conf. on Coastal Engineering, Vancouver, 653-668.

Noda, E. K., C. J. Sonu, V. C. Rupert, and J. I. Collins (1974), "Nearshore Circulation under Sea Breeze Conditions and Wave-Current Interaction in the Surf Zone," Tetra Tech Report TC-149-4.

Paddock, R. A., and J. D. Ditmars (1981), "Investigation of the Feasibility of Linking a Sediment Transport Model with a Nearshore Circulation Model," Report NUREG/CR-2237, ANL/EES-TM-155, Argonne National Laboratory, Argonne, Illinois.

Pawka, S. (1980), personal communication.

Pearce, B. R. (1972), "Numerical Calculation of the Response of Coastal Waters to Storm Systems with Applications to Hurricane Camille of August 17-22, 1969," College of Engineering Tech. Rept. 12, University of Florida.

Phillips, O. M. (1977), The Dynamics of the Upper Ocean, 2nd Ed., Cambridge University Press.

Roache, P. J. (1976), Computational Fluid Dynamics, Hermosa Publishers, Albuquerque.

Shore Protection Manual (1975), U.S. Army Corps of Engineers, Vol. 1, Government Printing Office.

Sonu, C. J. (1972), "Field Observations of Nearshore Circulation and Meandering Currents," J. Geophys. Res. 77(18), 3232-3247.

Sommerfeld, A. (1949), Partial Differential Equations in Physics, Academic Press, New York.

Symonds, G., D. A. Huntley, and A. J. Bowen (1982), "Two Dimensional Surf Beat: Long Wave Generation by a Time-Varying Breakpoint," J. Geophys. Res. 87(C1), 492-498.

Van Dorn, W. C. (1953), "Wind Stress on an Artificial Pond," J. Marine Res. 12,

Weggel, J. R. (1972), "Maximum Breaker Height," J. Waterways, Harbors and Coastal Engineering, ASCE 98, WW4, 529-548.

Wilson, B. W. (1966), "Seiche," in Encyclopedia of Oceanography, Reinhold Publishers, New York.

Wu, J. (1968), "Laboratory Studies of Wind-Wave Interactions," J. Fluid Mech. 34(1), 91-111.

Appendix I.

USING THE NEARSHORE CIRCULATION PROGRAM

The nearshore circulation model as described above is supplied in two versions, referred to as the linear and the nonlinear versions. Complete listings of the programs for each version follow. The programs as written are designed to run on a Burroughs B7700. The line

$RESET FREE

which appears on the front of each program allows for the use of standard Fortran on the B7700. This line should be removed before operating the program on a different system.

The two versions of the model are currently designed to be indistinguishable to the operator, with the input file containing wave and wind information and the depth grid being of identical format for each program. The output file generated currently at a line printer is also identical for each program. The exception to the uniformity between the two models occurs in the output file stored at the end of execution for subsequent restart of the model. Since the nonlinear version of the program requires the storage of three time levels of data, two extra time levels are included in the output. Thus, the nonlinear version of the program can not be started based on intermediate results generated by the linear version, as not enough information is present. It is possible in principle to start the linear version of the program using intermediate results from the non-linear version, although at present the READ statement in the linear version is not structured to handle this option.

The circulation model is currently designed to be run as a batch job, and as such has all device number specifications for data files in the job file external to the program. This feature may have to be changed depending on the machine to be used. Currently, data files may be named arbitrarily. The program currently requirues four IO device specifications:

Logical Device Number	Corresponding Data File
5	User defined input data
6	Output stream, currently directed to a line printer
3	Output file, stores results of last interation for a later restart
8	Input file used to restart the model

It should be noted that the input file 8 used for restarting the program is identical to, and should be just a renamed version of, the output file 3 generated by the model at the point from which it is desired to continue computation.

Structure of Input Data File 5

The input data file is structured into five groups of data. The first four groups each consist of a single line in the file, with data values entered unformatted; i.e., separated by commas with no requirements on spacing. The fifth data group consists of the depth grid, and requires more space than a single line, as explained below. The information included in the input data file is as follows.

DATA IN INPUT DATA FILE 5

1. Wave Parameters

   T - Wave period (seconds)

   HO - Wave height (meters)

   A - Wave angle, measured clockwise from the +X direction (degrees)

2. Wind Parameters

   WIND - Wind speed (meters/second)

   WINANG - Wind angle, measured clockwise from the -X direction (degrees)

3. Grid Parameters

   M - Number of grid rows in X direction (offshore)

   N - Number of grid columns in Y direction (longshore)

   DX - Grid spacing in X direction (meters)

   DY - Grid spacing in Y direction (meters)

   DT - Size of the time step (seconds)

   INDEX - Specifies the input of depth information
   =1, read data from previous run from file 8
   =2, read depth grid from input file 5
   =3, establish plane beach based on input beach slope

   AM - Beach slope (space in input file must be filled, but value is unsed only if INDEX=3)

4. Program Control Parameters

   ITA - Total number of iterations (including the accumulated total from previous runs, if previous run data is used)

   NHIGHT - Number of iterations over which the deep water wave builds up

ID - Determine if dissipation of wave energy is to be included
=0, no dissipation
=1, dissipation due to viscosity and bottom permeability is included

KSKIP - Determine frequency of printed iterations in output file 6

5. Depth Grid (required if INDEX=2)

D - Local water depth with respect to still water level at each grid center.

The depth grid D is input in an unformatted string of length M*N, starting with the fifth line of the input file 5 and continuing to as many lines as necessary. The data values are read off of an established grid row-wise in the +Y direction, starting at the inshore end of the grid.

Coordinate System Used in the Program

The input depth grid and wave and wind parameters should be established using the coordinates shown in Figure 2-1. The models require that the bathymetry be periodic in the longshore direction. Accordingly, for an input depth grid of size (M, N), where (I, J) are X and Y indices and (M, N) are the maximum values of (I, J), the laterally bordering depth values should satisfy the requirement:

$$D(I,N) = D(I,1) \quad .$$

If this condition is not met on input, D(I,1) is redefined to be equal to D(I,N). Both models create two additional rows in the Y direction, (N+1) and (N+2). The grid storage requirement for all variables is thus M by (N+2), with a current

maximum of 50 by 50. The maximum may be enlarged or decreased by alteration of all COMMON and DIMENSION statements in either program.

Wave angle and wind angle are defined as shown in Figure 2-1. Wave angle is measured clockwise from the +X direction. Waves approaching the beach from the right of directly offshore have wave angles > 180°. Wind angle is given clockwise from the -X direction, so that winds approaching shore from the right of normal have wind directions greater than 0°.

The program requires that incident waves have longshore components propagating in the +Y direction. For the case of wave angles = 180° - θ, the program flips the depth grid over and rotates the wave and wind directions through 360° - 2θ degrees, and runs the program for waves approaching at 180° + θ and a mirror image depth grid.

Error Messages from the Circulation Models

Each version of the nearshore circulation model will generate various error messages when fatal conditions are met during execution. The messages are, for the most part, identical from either program and correspond to equivalent situations within the programs. A list of possible internally occurring errors leading to the printing of a message follows.

1. Failure of iteration scheme to converge on a wave height at a grid point will cause the message

RELAXATION FOR THE WAVE HEIGHT FAILED TO CONVERGE

followed by an indication of the row and number of iterations tried. This

condition arises in the subroutine HEIGHT. Occurrence of the condition does not lead to termination of the program; iteration continues with the last wave height calculated and the entire iteration is tried again at the next time step.

2. Failure of iteration scheme to converge on a wave angle at a grid point will cause the message

RELAXATION FOR THETA FAILED AFTER -- ITERATIONS

with an indication given of the number of iterations tried. This condition could arise if the magnitude of a current directed against the general direction of wave travel was too strong. The condition arises in the subroutine ANGLE and causes termination of the program.

3. If a negative wave number RK is calculated, the message

RK IS NEGATIVE

is printed, and the program is terminated. This condition occurs in the subroutine WVNUM.

LINEAR MODEL

```
10000     $RESET FREE                                                                  00010000
10010     C*-------------------------------------------------------------------------  00010010
10020     C*                                                                           00010020
10030     C*    NEARSHORE CIRCULATION MODEL --- LINEAR VERSION                         00010030
10040     C*                                                                           00010040
10050     C*    THIS COMPUTER PROGRAM IS A NUMERICAL MODEL TO PREDICT SURF-ZONE        00010050
10060     C*    DYNAMICS, GIVEN A FIXED, PERIODIC TOPOGRAPHY AND DEEPWATER             00010060
10070     C*    MONOCHROMATIC WAVE CONDITIONS AS INPUT.                                00010070
10080     C*    THE REFRACTION PROGRAM INCLUDING WAVE-CURRENT INTERACTION IS BASED      00010080
10090     C*    ON THE PROGRAM OF NODA ET AL(TETRA TECH 1974).  THE CURRENT *          00010090
10100     C*    PROGRAM IS AN UPDATED VERSION OF THE MODEL DEVELOPED BY                00010100
10110     C*    BIRKEMEIER AND DALRYMPLE(UNIV. OF DEL. 1977).                          00010110
10120     C*                                                                           00010120
10130     C*-------------------------------------------------------------------------  00010130
10140           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),       00010140
10150          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),      00010150
10160          *DDDY(50,50),W(50,50),Y(50,50)                                         00010160
10170           COMMON/VAL/ETA(50,50)                                                 00010170
10180           COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50),    00010180
10190          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                                00010190
10200           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                           00010200
10210           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,              00010210
10220          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                             00010220
10230           DIMENSION DST(50,50)                                                  00010230
10240     C*-------------------------------------------------------------------------  00010240
10250     C*                                                                           00010250
10260     C*    NOTES ON RUNNING THE CIRCULATION PROGRAM                               00010260
10270     C*                                                                           00010270
10280     C*    1. WAVE ANGLE IS MEASURED CLOCKWISE FROM THE +X DIRECTION              00010280
10290     C*                                                                           00010290
10300     C*    2. WIND ANGLE IS MEASURED CLOCKWISE FROM THE -X DIRECTION              00010300
10310     C*                                                                           00010310
10320     C*    3. ALL INPUT AND OUTPUT DATA IS IN MKS UNITS                           00010320
10330     C*                                                                           00010330
10340     C*    4. THE DEPTH GRID IS OFFSET BY -DY FROM ALL THE OTHER VARIABLE         00010340
10350     C*       FIELDS                                                              00010350
10360     C*                                                                           00010360
10370     C*-------------------------------------------------------------------------  00010370
10380     C*                                                                           00010380
10390     C*    DEFINITIONS OF QUANTITIES USED IN PROGRAM                              00010390
10400     C*                                                                           00010400
10410     C*    1. CONSTANTS                                                           00010410
10420     C*                                                                           00010420
10430     C*       CF - BOTTOM FRICTION FACTOR (DARCY TYPE)                            00010430
10440     C*       VANCON - WIND SHEAR FACTOR (VAN DORN TYPE)                          00010440
10450     C*                                                                           00010450
10460     C*    2. VARIABLE ARRAYS (VALUE AT EACH GRID LOCATION)                       00010460
10470     C*                                                                           00010470
10480     C*       D - TOTAL WATER DEPTH (D(SW)+ETA)                                   00010480
10490     C*       ETA - DEVIATION OF MEAN WATER LEVEL FROM STILL WATER LEVEL          00010490
10500     C*       Z - WAVE ANGLE                                                      00010500
10510     C*       CO - COS(Z)                                                         00010510
10520     C*       SI - SIN(Z)                                                         00010520
10530     C*       H - WAVE HEIGHT                                                     00010530
10540     C*       HBREAK - LOCAL BREAKING WAVE HEIGHT                                 00010540
10550     C*       IB - BREAKING INDEX                                                 00010550
10560     C*            IB=0. WAVE IS BREAKING LOCALLY                                 00010560
10570     C*            IB=1. WAVE IS NOT BREAKING LOCALLY                             00010570
10580     C*       DDDX,DDDY - LOCAL DERIVATIVES OF THE TOTAL DEPTH                    00010580
```

```
10590    C*       CG - GROUP VELOCITY                                                    00010590
10600    C*       RKA - WAVE NUMBER                                                      00010600
10610    C*       U,V - X,Y VELOCITIES AT SIDES OF GRID BLOCKS                           00010610
10620    C*       W,Y - X,Y VELOCITIES AT CENTERS OF GRID BLOCKS                         00010620
10630    C*             NOTE THAT THE U,V AND W,Y ARRAYS ARE INTERCHANGED                00010630
10640    C*             IN SEVERAL SUBROUTINES THROUGH THE COMMON STATEMENTS             00010640
10650    C*       SIGXX,SIGXY,SIGYY - RADIATION STRESSES                                 00010650
10660    C*       TAUBX,TAUBY - BOTTOM STRESSES                                          00010660
10670    C*       TAUSX,TAUSY - SURFACE STRESSES                                         00010670
10680    C*                                                                              00010680
10690    C*    3. LOCALLY DEFINED VARIABLES                                              00010690
10700    C*                                                                              00010700
10710    C*       DUDX,DVDX - VELOCITY GRADIENT COMPONANTS                               00010710
10720    C*       DCDX,DCDY - WAVE CELERITY GRADIENT COMPONANTS                          00010720
10730    C*       DCGDX,DCGDY - GROUP VELOCITY GRADIENT COMPONANTS                       00010730
10740    C*       DTDX,DTDY - WAVE ANGLE GRADIENT COMPONANTS                             00010740
10750    C*       EPS - ACCURACY VALUE USED IN THE RELAXATION SCHEMES                    00010750
10760    C*                                                                              00010760
10770    C*                                                                              00010770
10780    C*                                                                              00010780
10790    C*--------------------------------------------------------------------------   00010790
10800    C*                                                                              00010800
10810    C*    VARIABLES TO BE READ INTO PROGRAM                                         00010810
10820    C*                                                                              00010820
10830    C*    1. WAVE PARAMETERS                                                        00010830
10840    C*                                                                              00010840
10850    C*       T       - WAVE PERIOD (SECONDS)                                        00010850
10860    C*       HO      - WAVE HEIGHT (METERS)                                         00010860
10870    C*       A       - WAVE ANGLE, CLOCKWISE FROM +X (DEGREES)                      00010870
10880    C*                                                                              00010880
10890    C*    2. WIND PARAMETERS                                                        00010890
10900    C*                                                                              00010900
10910    C*       WIND    - WIND SPEED (METERS/SECOND)                                   00010910
10920    C*       WINANG - WIND ANGLE, CLOCKWISE FROM -X (DEGREES)                       00010920
10930    C*                                                                              00010930
10940    C*    3. GRID PARAMETERS                                                        00010940
10950    C*                                                                              00010950
10960    C*       M       - NUMBER OF GRIDS IN X DIRECTION (OFFSHORE)                    00010960
10970    C*       N       - NUMBER OF GRIDS IN Y DIRECTION (LONGSHORE)                   00010970
10980    C*       DX      - GRID SPACING IN X DIRECTION (METERS)                         00010980
10990    C*       DY      - GRID SPACING IN Y DIRECTION (METERS)                         00010990
11000    C*       DT      - TIME STEP<DX/SQRT(2*G*DMAX) (SECONDS)                        00011000
11010    C*       INDEX   - SPECIFY INPUT OF DEPTH INFORMATON                            00011010
11020    C*                 =1, READ DATA FROM PREVIOUS RUN                              00011020
11030    C*                 =2, READ DEPTH GRID FROM INPUT DATA FILE                     00011030
11040    C*                 =3, ESTABLISH PLANE BEACH BASED ON INPUT BEACH SLOPE         00011040
11050    C*       AM      - BEACH SLOPE (REQUIRED IF INDEX=3)                            00011050
11060    C*                                                                              00011060
11070    C*    4. PROGRAM CONTROL PARAMETERS                                             00011070
11080    C*                                                                              00011080
11090    C*       ITA     - TOTAL NUMBER OF ITERATIONS (INCLUDING ACCUMULATED            00011090
11100    C*                 TOTAL OF PREVIOUS RUNS, IF PREVIOUS RUN DATA IS              00011100
11110    C*                 USED)                                                        00011110
11120    C*       NHIGHT - NUMBER OF ITERATIONS OVER WHICH THE DEEPWATER WAVE           00011120
11130    C*                 HEIGHT IS BUILT UP                                           00011130
11140    C*       ID      - DETERMINE IF DISSAPATION OF WAVE ENERGY IS TO BE             00011140
11150    C*                 INCLUDED                                                     00011150
11160    C*                 =0, NO DISSIPATION                                           00011160
11170    C*                 =1, DISSIPATION DUE TO VISCOSITY AND BOTTOM PERM-            00011170
11180    C*                     EABILITY                                                 00011180
11190    C*       KSKIP   - DETERMINE FREQUENCY OF PRINTED ITERATIONS                    00011190
11200    C*                                                                              00011200
```

```
11210   C*    5. DEPTH GRID (IF INDEX=2)                                        00011210
11220   C*                                                                      00011220
11230   C*       D        - LOCAL WATER DEPTH W/R MEAN WATER LEVEL.             00011230
11240   C*                  SEE USERS MANUAL FOR INPUT FORMAT                   00011240
11250   C*                                                                      00011250
11260   C*--------------------------------------------------------------------- 00011260
11270   C*                                                                      00011270
11280   C*    DEFINE CONSTANTS                                                  00011280
11290   C*--------------------------------------------------------------------- 00011290
11300         RHO=1000.                                                         00011300
11310         G=9.8                                                             00011310
11320         PI=3.1415927                                                      00011320
11330         PI2=2.*PI                                                         00011330
11340         RAD=180.0/PI                                                      00011340
11350         CF=0.015                                                          00011350
11360         DD=0.0                                                            00011360
11370   C*--------------------------------------------------------------------- 00011370
11380   C*    READ INPUT DATA                                                   00011380
11390   C*                                                                      00011390
11400   C*    1. WAVE PARAMETERS                                                00011400
11410         READ(5,/) T,HO,A                                                  00011410
11420   C*    2. WIND PARAMETERS                                                00011420
11430         READ(5,/) WIND,WINANG                                             00011430
11440   C*    3. GRID PARAMETERS                                                00011440
11450         READ(5,/) M,N,DX,DY,DT,INDEX,AM                                   00011450
11460   C*    4. PROGRAM CONTROL PARAMETERS                                     00011460
11470         READ(5,/) ITA,NHIGHT,ID,KSKIP                                     00011470
11480   C*--------------------------------------------------------------------- 00011480
11490   C*    WRITE INPUT DATA ON OUTPUT FILE AS HEADER                         00011490
11500   C*--------------------------------------------------------------------- 00011500
11510         WRITE(6,102)M,N,ITA                                               00011510
11520         WRITE(6,101)DX,DY,DT                                              00011520
11530         WRITE(6,119) T,HO,A,AM                                            00011530
11540         WRITE(6,114) WIND,WINANG                                          00011540
11550         WRITE(6,115) INDEX,ID                                             00011550
11560         WRITE(6,117) NHIGHT                                               00011560
11570         WRITE(6,116) CF,DD                                                00011570
11580         ITO=0                                                             00011580
11590         HEIGHT=HO                                                         00011590
11600         DELTAT=DT                                                         00011600
11610         WINANG=WINANG/RAD                                                 00011610
11620         N2=N+2                                                            00011620
11630         N1=N+1                                                            00011630
11640         M2=M-2                                                            00011640
11650         M1=M-1                                                            00011650
11660         DX2=DX*2.                                                         00011660
11670         DY2=DY*2.                                                         00011670
11680         SIGMA=PI2/T                                                       00011680
11690         EPS=0.01                                                          00011690
11700   C*--------------------------------------------------------------------- 00011700
11710   C*    ESTABLISH BOTTOM TOPOGRAPHY BASED ON VALUE OF INDEX               00011710
11720   C*--------------------------------------------------------------------- 00011720
11730         GO TO (1,2,3) INDEX                                               00011730
11740   C*--------------------------------------------------------------------- 00011740
11750   C*    INDEX=1, READ DATA FROM FILE 8 FOR PREVIOUS RUN OF MODEL          00011750
11760   C*--------------------------------------------------------------------- 00011760
11770     1   READ(8,113)                                                       00011770
11780        *((ETA(I,J),J=1,N2),I=1,M),((D(I,J),J=1,N2),I=1,M),               00011780
11790        *((U(I,J),J=1,N2),I=1,M),((V(I,J),J=1,N2),I=1,M),                 00011790
11800        *((Z(I,J),J=1,N2),I=1,M),((H(I,J),J=1,N2),I=1,M),                 00011800
11810        *((W(I,J),J=1,N2),I=1,M),((Y(I,J),J=1,N2),I=1,M),                 00011810
11820        *((RKA(I,J),J=1,N2),I=1,M)                                        00011820
```

```
11830          READ(8,107)ITO                                                  00011830
11840          DO 10 I=1,M                                                     00011840
11850          DO 10 J=1,N2                                                    00011850
11860          CO(I,J)=COS(Z(I,J))                                             00011860
11870          SI(I,J)=SIN(Z(I,J))                                             00011870
11880       10 CONTINUE                                                        00011880
11890     C*-------------------------------------------------------------------00011890
11900     C*    IF WAVE ANGLE IS LESS THAN 180 DEGREES, VARIABLE FIELDS WERE  00011900
11910     C*    FLIPPED DURING PREVIOUS RUN.  ROTATE THE WAVE AND WIND ANGLES 00011910
11920     C*-------------------------------------------------------------------00011920
11930          IF(A.GE.180.) GO TO 42                                          00011930
11940          A=360.-A                                                        00011940
11950          WINANG=PI2-WINANG                                               00011950
11960          GO TO 42                                                        00011960
11970     C*-------------------------------------------------------------------00011970
11980     C*    INDEX=2, READ IN DEPTH GRID FROM INPUT DATA FILE 5            00011980
11990     C*-------------------------------------------------------------------00011990
12000        2  READ(5,/,END=99)((D(I,J),J=1,N),I=1,M)                         00012000
12010       99  WRITE(6,120) I,J                                               00012010
12020     C*-------------------------------------------------------------------00012020
12030     C*    IF WAVE ANGLE IS LESS THAN 180 DEG., FLIP THE INPUT DEPTH     00012030
12040     C*    GRID AND ROTATE THE WAVE AND WIND ANGLES                      00012040
12050     C*-------------------------------------------------------------------00012050
12060          IF(A.GT.180.) GO TO 22                                          00012060
12070          A=360.-A                                                        00012070
12080          WINANG=PI2-WINANG                                               00012080
12090          DO 21 I=1,M                                                     00012090
12100          DO 20 J=1,N                                                     00012100
12110       20 DST(I,J)=D(I,J)                                                 00012110
12120          DO 21 J=1,N                                                     00012120
12130          K=(N-J)+1                                                       00012130
12140       21 D(I,K)=DST(I,J)                                                 00012140
12150       22 DO 23 I=1,M                                                     00012150
12160          D(I,1)=D(I,N)                                                   00012160
12170          D(I,N2)=D(I,3)                                                  00012170
12180          D(I,N1)=D(I,2)                                                  00012180
12190       23 CONTINUE                                                        00012190
12200          GO TO 41                                                        00012200
12210     C*-------------------------------------------------------------------00012210
12220     C*    INDEX=3, ESTABLISH PLANE BEACH BASED ON VALUE OF AM READ ABOVE 00012220
12230     C*-------------------------------------------------------------------00012230
12240        3  DO 31 J=1,N2                                                   00012240
12250          DO 31 I=1,M                                                     00012250
12260          D(I,J)=((FLOAT(I-2)*DX)*AM)                                     00012260
12270       31 CONTINUE                                                        00012270
12280     C*-------------------------------------------------------------------00012280
12290     C*    IF WAVE ANGLE IS LESS THAN 180 DEGREES, ROTATE THE WAVE AND WIND 00012290
12300     C*    ANGLES                                                        00012300
12310     C*-------------------------------------------------------------------00012310
12320          IF(A.GE.180.) GO TO 41                                          00012320
12330          A=360.-A                                                        00012330
12340          WINANG=PI2-WINANG                                               00012340
12350       41 WRITE(6,108)                                                    00012350
12360          WRITE(6,103)((D(I,J),J=1,N1),I=1,M)                             00012360
12370          DO 65 I=1,M                                                     00012370
12380          DO 65 J=1,N2                                                    00012380
12390          ETA(I,J)=0.0                                                    00012390
12400          U(I,J)=0.0                                                      00012400
12410          V(I,J)=0.0                                                      00012410
12420       65 CONTINUE                                                        00012420
12430       42 DO 15 I=1,M                                                     00012430
12440          IF(D(I,1).GT.DD) GO TO 95                                       00012440
```

```
12450        15 CONTINUE                                                    00012450
12460        95 IWET=I                                                      00012460
12470           IDRY=IWET-1                                                 00012470
12480     C*-------------------------------------------------------------- 00012480
12490     C*    MAIN PROGRAM LOOP FOR EACH ITERATION                        00012490
12500     C*-------------------------------------------------------------- 00012500
12510           DO 4 IT=(1+ITO),ITA                                         00012510
12520           L=AMOD(FLOAT(IT),FLOAT(KSKIP))                              00012520
12530           HO=HEIGHT*TANH(FLOAT(IT)/(FLOAT(NHIGHT)/2.0))               00012530
12540           IF(IT.LE.(NHIGHT*2.0)) GO TO 12                             00012540
12550           GO TO 13                                                    00012550
12560        12 CALL DGRAD                                                  00012560
12570           CALL REFRAC(A,HO,IT,INDEX,NHIGHT,CF)                        00012570
12580        13 CALL TAUSB(WIND,WINANG,CF,ITO)                              00012580
12590           CALL UCALC                                                  00012590
12600           CALL ETAS                                                   00012600
12610           SUM=0.0                                                     00012610
12620           DO 86 I=1,M                                                 00012620
12630           DO 86 J=2,N                                                 00012630
12640           SUM=SUM+0.5*(W(I,J)**2+V(I,J)**2)*D(I,J-1)                  00012640
12650           SUM=SUM+G*ETA(I,J)*(D(I,J-1)-0.5*ETA(I,J))                  00012650
12660        86 CONTINUE                                                    00012660
12670     C*-------------------------------------------------------------- 00012670
12680     C*    WRITE SUM AFTER EACH ITERATION                              00012680
12690     C*-------------------------------------------------------------- 00012690
12700           WRITE(6,100) SUM                                            00012700
12710           IF(L.NE.0)GO TO 4                                           00012710
12720     C*-------------------------------------------------------------- 00012720
12730     C*    WRITE WAVE HEIGHT, WAVE ANGLE, VELOCITIES, AND SETUP AT     00012730
12740     C*    KSKIP INTERVALS                                             00012740
12750     C*-------------------------------------------------------------- 00012750
12760           WRITE(6,109)                                                00012760
12770           WRITE(6,110)IT,HO                                           00012770
12780           WRITE(6,111)                                                00012780
12790           WRITE(6,112)                                                00012790
12800           WRITE(6,103)((H(I,J),J=1,N1),I=1,M)                         00012800
12810           DO 401 I=1,M                                                00012810
12820           DO 401 J=1,N1                                               00012820
12830       401 ZZ(I,J)=360.0-Z(I,J)*RAD                                    00012830
12840           WRITE(6,118)                                                00012840
12850           WRITE(6,103)((ZZ(I,J),J=1,N1),I=1,M)                        00012850
12860           WRITE (6,104)                                               00012860
12870           WRITE (6,103) ((U(I,J), J = 1,N1),I=1,M)                    00012870
12880           WRITE (6,105)                                               00012880
12890           WRITE (6,103) ((V(I,J), J = 1,N1),I=1,M)                    00012890
12900           WRITE (6,106)                                               00012900
12910           WRITE(6,103) ((ETA(I,J),J=1,N1),I=1,M)                      00012910
12920       4   CONTINUE                                                    00012920
12930     C*-------------------------------------------------------------- 00012930
12940     C*    WRITE THE RESULTS OF THE LAST ITERATION INTO AN OUTPUT FILE 00012940
12950     C*    TO BE USED FOR A SUBSEQUENT STARTUP                         00012950
12960     C*-------------------------------------------------------------- 00012960
12970           WRITE(3,113)                                                00012970
12980          *((ETA(I,J),J=1,N2),I=1,M),((D(I,J),J=1,N2),I=1,M),          00012980
12990          *((U(I,J),J=1,N2),I=1,M),((V(I,J),J=1,N2),I=1,M),            00012990
13000          *((Z(I,J),J=1,N2),I=1,M),((H(I,J),J=1,N2),I=1,M),            00013000
13010          *((W(I,J),J=1,N2),I=1,M),((Y(I,J),J=1,N2),I=1,M)             00013010
13020          *,((RKA(I,J),J=1,N2),I=1,M)                                  00013020
13030           WRITE(3,107)ITA                                             00013030
13040           LOCK 3                                                      00013040
13050       100 FORMAT(1X,"THE TOTAL ENERGY IS",F14.2)                      00013050
13060       101 FORMAT (/,1X,'DX=',I5,3X,'DY=',I5,3X,3X,'DT=',F5.2)         00013060
```

```
13070      102 FORMAT(1X,"M=",I5,3X,"N=",I5,3X,"ITERATIONS =",I5)                00013070
13080      103 FORMAT(1X,11F10.4)                                                00013080
13090      104 FORMAT ( 10X,12HX - VELOCITY)                                     00013090
13100      105 FORMAT ( 10X,12HY - VELOCITY)                                     00013100
13110      106 FORMAT (10X,10HETA VALUES)                                        00013110
13120      107 FORMAT(I5)                                                        00013120
13130      108 FORMAT (//' THE AUGMENTED MATRIX OF WATER DEPTH IN METERS'/)      00013130
13140      109 FORMAT('*****************************',/)                        00013140
13150      110 FORMAT(10X,'ITERATION NUMBER',I6,10X,'WAVE HEIGHT=',F10.5,/)      00013150
13160      111 FORMAT (1X,'*****************************',/)                     00013160
13170      112 FORMAT (5X,'WAVE HEIGHTS',/)                                      00013170
13180      113 FORMAT(15F8.4)                                                    00013180
13190      114 FORMAT (1X,'WIND SPEED =',F7.3,'M/SEC',5X,'WIND ANGLE =',F7.      00013190
13200         *3,' DEGREES')                                                     00013200
13210      115 FORMAT(5X,'INDEX =',I5,3X,'ID =',I5)                              00013210
13220      116 FORMAT(10X,4HCF =,F10.3,5X,4HDD =,F10.3)                          00013220
13230      117 FORMAT(10X,'WAVE HEIGHT BUILDS UP FIRST',I3,' ITERATIONS')        00013230
13240      118 FORMAT(2X,'WAVE ANGLE (DEGREES)')                                 00013240
13250      119 FORMAT(1X,'WAVE PARAMETERS',/2X,'PERIOD =',F7.2,'  HEIGHT =',F7   00013250
13260         *.2,3X,7HANGLE =,F7.2,1X,7HDEGREES,4X,12HBEACH SLOPE=,F7.4)        00013260
13270      120  FORMAT(" INPUT DEPTH GRID TOO SMALL. LAST I,J VALUES ARE"/       00013270
13280         12X,I2,2X,I2)                                                      00013280
13290           STOP                                                            00013290
13300           END                                                             00013300
13310     C*--------------------------------------------------------------------00013310
13320     C*                                                                    00013320
13330           SUBROUTINE REFRAC(THETAO,HH,ITER,INDEX,NHIGHT,CF)               00013330
13340     C*                                                                    00013340
13350     C*    REFRAC CONTROLS THE CALCULATION OF WAVE ANGLE AND WAVE HEIGHT   00013350
13360     C*    FOR EACH ITERATION                                              00013360
13370     C*                                                                    00013370
13380     C*--------------------------------------------------------------------00013380
13390           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50), 00013390
13400          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00013400
13410          *,DDDY(50,50)                                                    00013410
13420           COMMON/STRESS/SIGX(50,50),SIGY(50,50),SIGXY(50,50),TAUBX(50,50), 00013420
13430          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                          00013430
13440           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,        00013440
13450          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00013450
13460           IF (ITER.GT.(NHIGHT)) GO TO 600                                 00013460
13470     C*--------------------------------------------------------------------00013470
13480     C*    CALL SNELL DURING BUILDUP OF DEEP WATER WAVE                    00013480
13490     C*--------------------------------------------------------------------00013490
13500           THETA=PI2-THETAO/RAD                                            00013500
13510           CALL SNELL (THETA,HH,ITER)                                      00013510
13520       600 DO 50 I=1,IDRY                                                  00013520
13530           DO 50 J=1,N2                                                    00013530
13540           H(I,J)=0.0                                                      00013540
13550           Z(I,J)=PI                                                       00013550
13560           SI(I,J)=0.0                                                     00013560
13570           CO(I,J)=-1.0                                                    00013570
13580        50 CONTINUE                                                        00013580
13590           CALL ANGLE(25)                                                  00013590
13600           CALL HEIGHT(25,CF)                                              00013600
13610     C*--------------------------------------------------------------------00013610
13620     C*    CALCULATE THE RADIATION STRESSES                                00013620
13630     C*--------------------------------------------------------------------00013630
13640           DO 57 I=IWET,M1                                                 00013640
13650           DO 21 J=3,N1                                                    00013650
13660           ENERGY=0.125*RHO*G*(H(I,J)**2)                                  00013660
13670           SIGX(I,J)=SIGX(I,J)*ENERGY                                      00013670
13680           SIGY(I,J)=SIGY(I,J)*ENERGY                                      00013680
```

```
13690          SIGXY(I,J)=SIGXY(I,J)*ENERGY                                    00013690
13700       21 CONTINUE                                                        00013700
13710          SIGX(I,1)=SIGX(I,N)                                             00013710
13720          SIGY(I,1)=SIGY(I,N)                                             00013720
13730          SIGXY(I,1)=SIGXY(I,N)                                           00013730
13740          SIGX(I,2)=SIGX(I,N1)                                            00013740
13750          SIGY(I,2)=SIGY(I,N1)                                            00013750
13760          SIGXY(I,2)=SIGXY(I,N1)                                          00013760
13770          SIGX(I,N2)=SIGX(I,3)                                            00013770
13780          SIGY(I,N2)=SIGY(I,3)                                            00013780
13790          SIGXY(I,N2)=SIGXY(I,3)                                          00013790
13800       57 CONTINUE                                                        00013800
13810          DO 59 J=3,N1                                                    00013810
13820          SIGXY(M,J)=2.0*SIGXY(M1,J)-SIGXY(M2,J)                          00013820
13830       59 CONTINUE                                                        00013830
13840          SIGXY(M,1)=SIGXY(M,N)                                           00013840
13850          SIGXY(M,2)=SIGXY(M,N1)                                          00013850
13860          SIGXY(M,N2)=SIGXY(M,3)                                          00013860
13870          RETURN                                                          00013870
13880          END                                                             00013880
13890    C*--------------------------------------------------------------------00013890
13900    C*                                                                    00013900
13910          SUBROUTINE GROUP(I,J,DCGDX,DCGDY,FF)                            00013910
13920    C*                                                                    00013920
13930    C*     GROUP CALCULATES THE GROUP VELOCITY, WAVE CELERITY, AND        00013930
13940    C*     THEIR SPATIAL DERIVATIVES                                      00013940
13950    C*                                                                    00013950
13960    C*--------------------------------------------------------------------00013960
13970          COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),  00013970
13980         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00013980
13990         *,DDDY(50,50),U(50,50),V(50,50)                                  00013990
14000          COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,        00014000
14010         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00014010
14020          COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                     00014020
14030    C*--------------------------------------------------------------------00014030
14040    C*     STATEMENT FUNCTIONS DUDX,DUDY,DVDX,DVDY,DTDX,DTDY,E,DKDX,DKDY  00014040
14050    C*--------------------------------------------------------------------00014050
14060          DUDX(I,J)=(W(I+1,J)-W(I,J))/DX                                  00014060
14070          DUDY(I,J)=(U(I,J+1)-U(I,J-1))/DY2                               00014070
14080          DVDX(I,J)=(V(I+1,J)-V(I-1,J))/DX2                               00014080
14090          DVDY(I,J)=(Y(I,J+1)-Y(I,J))/DY                                  00014090
14100          DTDX(I,J)=(Z(I+1,J)-Z(I-1,J))/DX2                               00014100
14110          DTDY(I,J)=(Z(I,J+1)-Z(I,J-1))/DY2                               00014110
14120          E(I,J)=U(I,J)*COSI+V(I,J)*SINI+0.5*A*(1.0+ARG2/SINH2)/RK        00014120
14130          DKDX(I,J)=RK*((U(I,J)*SINI-V(I,J)*COSI)*DTDX(I,J)-(COSI*DUDX(I,J)+ 00014130
14140         *SINI*DVDX(I,J))-A*DDDX(I,J-1)/SINH2)/EE                         00014140
14150          DKDY(I,J)=RK*((U(I,J)*SINI-V(I,J)*COSI)*DTDY(I,J) -(COSI*DUDY(I,J) 00014150
14160         * +SINI*DVDY(I,J))-A*DDDY(I,J-1)/SINH2)/EE                       00014160
14170          JJ=J-1                                                          00014170
14180          DEP=D(I,JJ)                                                     00014180
14190          COSI=CO(I,J)                                                    00014190
14200          SINI=SI(I,J)                                                    00014200
14210          DCGDX=0.0                                                       00014210
14220          DCGDY=0.0                                                       00014220
14230    C*--------------------------------------------------------------------00014230
14240    C*     CHECK FOR DRY LAND                                             00014240
14250    C*-------------------------------------------------------------------*---00014250
14260          IF(DEP.LE.DD) RETURN                                            00014260
14270          CALL WVNUM(DEP,COSI,SINI,U(I,J),V(I,J),RK,A)                    00014270
14280          RKA(I,J) = RK                                                   00014280
14290          IF (J.EQ.N) RKA(I,1)=RKA(I,J)                                   00014290
14300          IF (J.EQ.2) RKA(I,N1)=RKA(I,J)                                  00014300
```

```
14310          IF (J.EQ.3) RKA(I,N2)=RKA(I,J)                                   00014310
14320          TA=TANH(RK*DEP)                                                  00014320
14330          HBREAK(I,J)=0.12*PI2*TA/RK                                       00014330
14340          COSH1=COSH(RK*DEP)                                               00014340
14350          SECHSQ=1.0/(COSH1**2)                                            00014350
14360          ARG2=2.0*RK*DEP                                                  00014360
14370          SINH2=SINH(ARG2)                                                 00014370
14380          COSH2=COSH(ARG2)                                                 00014380
14390          SINHSQ=SINH2**2                                                  00014390
14400          EE=E(I,J)                                                        00014400
14410          C=SQRT(G*TA/RK)                                                  00014410
14420          FF=0.5*(1.0+ARG2/SINH2)                                          00014420
14430          CG(I,J)=FF*C                                                     00014430
14440          P=C*(SINH2-ARG2*COSH2)/SINHSQ                                    00014440
14450          DKDDX=RK*DDDX(I,JJ)+DEP*DKDX(I,J)                                00014450
14460          DKDDY=RK*DDDY(I,JJ)+DEP*DKDY(I,J)                                00014460
14470          Q=0.5*G/(C*RK**2)                                                00014470
14480          DCDX=Q*(RK*SECHSQ*DKDDX-TA*DKDX(I,J))                            00014480
14490          DCDY=Q*(RK*SECHSQ*DKDDY-TA*DKDY(I,J))                            00014490
14500          DCGDX=P*DKDDX+FF*DCDX                                            00014500
14510          DCGDY=P*DKDDY+FF*DCDY                                            00014510
14520          RETURN                                                           00014520
14530          END                                                              00014530
14540    C*--------------------------------------------------------------------00014540
14550    C*                                                                     00014550
14560          SUBROUTINE HEIGHT(ITMAX,CF)                                      00014560
14570    C*                                                                     00014570
14580    C*    HEIGHT CALCULATES THE WAVE HEIGHT                                00014580
14590    C*                                                                     00014590
14600    C*--------------------------------------------------------------------00014600
14610          COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),  00014610
14620         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00014620
14630         *,DDDY(50,50),U(50,50),V(50,50)                                  00014630
14640          COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,         00014640
14650         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00014650
14660          COMMON/STRESS/SIGX(50,50),SIGY(50,50),SIGXY(50,50),TAUBX(50,50), 00014660
14670         *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                          00014670
14680          COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                      00014680
14690           VISCOS=1.0E-05                                                  00014690
14700           PERM=1.06E-09                                                   00014700
14710          DO 500 I=2,M1                                                    00014710
14720          DO 500 J=2,N1                                                    00014720
14730          IF(D(I,J-1).LE.DD) GO TO 499                                     00014730
14740          CALL GROUP(I,J,DCGDX,DCGDY,FF)                                   00014740
14750          DUDX=(W(I+1,J)-W(I,J))/DX                                        00014750
14760          DUDY=(U(I,J+1)-U(I,J-1))/DY2                                     00014760
14770          DVDX=(V(I+1,J)-V(I-1,J))/DX2                                     00014770
14780          DVDY=(Y(I,J+1)-Y(I,J))/DY                                        00014780
14790          DTDX=(Z(I+1,J)-Z(I-1,J))/DX2                                     00014790
14800          DTDY=(Z(I,J+1)-Z(I,J-1))/DY2                                     00014800
14810     501  SS2=SI(I,J)**2                                                   00014810
14820          CC2=CO(I,J)**2                                                   00014820
14830          CH=COSH(RKA(I,J)*D(I,J-1))                                       00014830
14840          IF(ID.EQ.1) GO TO 301                                            00014840
14850          X=0.0                                                            00014850
14860          GO TO 302                                                        00014860
14870     301  X=RKA(I,J)*CF*SIGMA*H(I,J)/(6.*PI*(CH**3.-CH))                  00014870
14880          X=X+G*PERM*RKA(I,J)/(VISCOS*CH*CH)                               00014880
14890     302  NA=N1                                                            00014890
14900          NB=N2                                                            00014900
14910          SIGXX=(2.0*FF-0.5)*CC2+(FF-0.5)*SS2                              00014910
14920          SIGYY=(2.0*FF-0.5)*SS2+(FF-0.5)*CC2                              00014920
```

```
14930          SIGX(I,J) = SIGXX                                                00014930
14940          SIGY(I,J) = SIGYY                                                00014940
14950          TAUXY=FF*SI(I,J)*CO(I,J)                                         00014950
14960          SIGXY(I,J)=TAUXY                                                 00014960
14970          S(I,J)=CG(I,J)*(SI(I,J)*DTDX-CO(I,J)*DTDY)-(DUDX+DVDY)-(CO(I,J)*DC 00014970
14980         *GDX+SI(I,J)*DCGDY)-(SIGXX*DUDX+TAUXY*DUDY+TAUXY*DVDX+SIGYY*DVDY)-X 00014980
14990          GO TO 500                                                        00014990
15000      499 H(I,J)=0.0                                                       00015000
15010          SIGXY(I,J)=0.0                                                   00015010
15020      500 CONTINUE                                                         00015020
15030          DO 510 II=1,M2                                                   00015030
15040          I=M-II                                                           00015040
15050          DO 580 ITT=1,ITMAX                                               00015050
15060          IFLAG=1                                                          00015060
15070          DO 520 J=2,N1                                                    00015070
15080          IF(D(I,J-1) .LE. DD) GO TO 520                                   00015080
15090          IB(I,J)=1                                                        00015090
15100          CC1=(V(I,J)+CG(I,J)*SI(I,J))/DY                                  00015100
15110          CC2=(U(I,J)+CG(I,J)*CO(I,J))/DX                                  00015110
15120          HNEW(I,J)= (CC1*H(I,J-1)-CC2*H(I+1,J))/(CC1-CC2-S(I,J)/2.0)      00015120
15130          IF (HNEW(I,J) .LT. 0.0) GO TO 810                                00015130
15140          IF(HNEW(I,J).LE.HBREAK(I,J)) GO TO 850                           00015140
15150      810 HNEW(I,J)=HBREAK(I,J)                                            00015150
15160          IB(I,J)=0                                                        00015160
15170      850 CONTINUE                                                         00015170
15180          IF(ABS(HNEW(I,J)-H(I,J)).GT.(EPS*ABS(HNEW(I,J)))) IFLAG=0        00015180
15190      C*--------------------------------------------------------------------00015190
15200      C*     BOUNDARY CONDITIONS                                           00015200
15210      C*--------------------------------------------------------------------00015210
15220          IF(J.NE.2) GO TO 800                                             00015220
15230          HNEW(I,N1)=HNEW(I,J)                                             00015230
15240          IB(I,N1)=IB(I,J)                                                 00015240
15250          GO TO 520                                                        00015250
15260      800 IF(J.NE.3) GO TO 801                                             00015260
15270          HNEW(I,N2)=HNEW(I,J)                                             00015270
15280          IB(I,N2)=IB(I,J)                                                 00015280
15290      801 IF(J.NE.N) GO TO 802                                             00015290
15300          HNEW(I,1)=HNEW(I,J)                                              00015300
15310          IB(I,1)=IB(I,J)                                                  00015310
15320          GO TO 520                                                        00015320
15330      802 IF(J.NE.N1) GO TO 520                                            00015330
15340          HNEW(I,2)=HNEW(I,J)                                              00015340
15350          IB(I,2)=IB(I,J)                                                  00015350
15360      520 CONTINUE                                                         00015360
15370          DO 625 J=1,N2                                                    00015370
15380          H(I,J)=HNEW(I,J)                                                 00015380
15390      625 CONTINUE                                                         00015390
15400          IF(IFLAG.EQ.1) GO TO 570                                         00015400
15410      580 CONTINUE                                                         00015410
15420          WRITE(6,540) I,ITT                                               00015420
15430      540 FORMAT(' RELAXATION FOR THE WAVE HEIGHT FAILED TO CONVERGE',     00015430
15440         *'ON ROW ',I5,'AFTER',I6,'ITERATIONS')                            00015440
15450          WRITE(6,541)(H(I,J),J=1,N2    )                                  00015450
15460      541 FORMAT('  LAST VALUES OF H ARE'/,10F13.5)                        00015460
15470      C       RETURN                                                       00015470
15480      570 CONTINUE                                                         00015480
15490      510 CONTINUE                                                         00015490
15500          RETURN                                                           00015500
15510          END                                                              00015510
15520      C*--------------------------------------------------------------------00015520
15530      C*                                                                  00015530
15540          SUBROUTINE ANGLE(ITMAX)                                          00015540
```

```
15550     C*                                                                    00015550
15560     C*     ANGLE CALCULATES THE LOCAL WAVE DIRECTION                      00015560
15570     C*                                                                    00015570
15580     C*--------------------------------------------------------------------00015580
15590           COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),  00015590
15600          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)  00015600
15610          *,DDDY(50,50),U(50,50),V(50,50)                                  00015610
15620           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,        00015620
15630          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00015630
15640     C*--------------------------------------------------------------------00015640
15650     C*     DEFINE STATEMENT FUNCTIONS - C,SS,DUDX,DUDY,DVDX,DVDY,DKDX,    00015650
15660     C*     DKDY,FAC,F                                                     00015660
15670     C*--------------------------------------------------------------------00015670
15680           C(I,J)=0.25*(CO(I+1,J)+CO(I-1,J)+CO(I,J+1)+CO(I,J-1))+0.125*((Z(I 00015680
15690          *+1,J)-Z(I-1,J))*(SI(I+1,J)-SI(I-1,J))+ (Z(I,J+1)-Z(I,J-1))*(SI(I, 00015690
15700          *J+1)-SI(I,J-1)))                                                00015700
15710           SS(I,J)=0.25*(SI(I+1,J)+SI(I-1,J)+SI(I,J+1)+SI(I,J-1))+0.125*((Z(I 00015710
15720          *+1,J)-Z(I-1,J))*(CO(I-1,J)-CO(I+1,J)) + (Z(I,J+1)-Z(I,J-1))*(CO(I, 00015720
15730          *J-1)-CO(I,J+1)))                                                00015730
15740           DUDX(I,J)=(W(I+1,J)-W(I,J))/DX                                  00015740
15750           DUDY(I,J)=(U(I,J+1)-U(I,J-1))/DY2                               00015750
15760           DVDX(I,J)=(V(I+1,J)-V(I-1,J))/DX2                               00015760
15770           DVDY(I,J)=(Y(I,J+1)-Y(I,J))/DY                                  00015770
15780           F(I,J)=U(I,J)*COSI+V(I,J)*SINI + 0.5*A*(1.0 + ARG2/SINH2)/RK    00015780
15790           DKDY(I,J)=(-(COSI*DUDY(I,J) + SINI*DVDY(I,J))-A*DDDY(I,J-1)/SINH2) 00015790
15800          */FF                                                             00015800
15810           DKDX(I,J)=(-(COSI*DUDX(I,J)+SINI*DVDX(I,J))-A*DDDX(I,J-1)/SINH2)/F 00015810
15820          *F                                                               00015820
15830           FAC(I,J)=U(I,J)*SINI-V(I,J)*COSI                                00015830
15840           DO 200 ITT=1,ITMAX                                              00015840
15850           MWET=M-IWET                                                     00015850
15860           IFLAG=1                                                         00015860
15870           DO 210 II=1,MWET                                                00015870
15880           I=M-II                                                          00015880
15890           DO 210 J=2,N1                                                   00015890
15900           IF(D(I,J-1).LE.DD) GO TO 210                                    00015900
15910           COSI=C(I,J)                                                     00015910
15920           SINI=SS(I,J)                                                    00015920
15930           JJ=J-1                                                          00015930
15940           CALL WVNUM(D(I,JJ),COSI,SINI,U(I,J),V(I,J),RK,A)                00015940
15950           ARG2=2.0*RK*D(I,JJ)                                             00015950
15960           SINH2=SINH(ARG2)                                                00015960
15970           FF=F(I,J)                                                       00015970
15980           IF(FF.GT.0.0) GO TO 450                                         00015980
15990           WRITE(6,451) I,J,D(I,JJ),COSI,SINI,U(I,J),V(I,J),RK,A           00015990
16000       451 FORMAT(10X'FF IS NEGATIVE--OUTPUT I,J,D,COSI,SINI,U,V,RK,A'/,   00016000
16010          *2I5,7E13.4)                                                     00016010
16020           GO TO 210                                                       00016020
16030       450 FACI=FAC(I,J)                                                   00016030
16040           DEN1=(SINI-COSI*FACI/FF)/DY                                     00016040
16050           DEN2=(COSI+SINI*FACI/FF)/DX                                     00016050
16060           DEN=DEN1-DEN2                                                   00016060
16070           ZNEW=(COSI*DKDY(I,J)-SINI*DKDX(I,J)+Z(I,J-1)*DEN1-Z(I+1,J)*     00016070
16080          *DEN2)/DEN                                                       00016080
16090           IF(ABS(ZNEW-Z(I,J)).GT.(EPS*ABS(ZNEW))) IFLAG=0                 00016090
16100           Z(I,J)=ZNEW                                                     00016100
16110           CO(I,J)=COS(Z(I,J))                                             00016110
16120           SI(I,J)=SIN(Z(I,J))                                             00016120
16130           IF(J.NE.2) GO TO 400                                            00016130
16140           Z(I,N1)=Z(I,2)                                                  00016140
16150           CO(I,N1)=CO(I,2)                                                00016150
16160           SI(I,N1)=SI(I,2)                                                00016160
```

```
16170          GO TO 210                                                    00016170
16180      400 IF(J.NE.3) GO TO 401                                         00016180
16190          Z(I,N2)=Z(I,3)                                               00016190
16200          CO(I,N2)=CO(I,3)                                             00016200
16210          SI(I,N2)=SI(I,3)                                             00016210
16220          GO TO 210                                                    00016220
16230      401 IF(J.NE.N) GO TO 402                                         00016230
16240          Z(I,1)=Z(I,N)                                                00016240
16250          CO(I,1)=CO(I,N)                                              00016250
16260          SI(I,1)=SI(I,N)                                              00016260
16270          GO TO 210                                                    00016270
16280      402 CONTINUE                                                     00016280
16290          IF(J.NE.N1) GO TO 210                                        00016290
16300          Z(I,2)=Z(I,N1)                                               00016300
16310          CO(I,2)=CO(I,N1)                                             00016310
16320          SI(I,2)=SI(I,N1)                                             00016320
16330      210 CONTINUE                                                     00016330
16340          IF(IFLAG.EQ.1) GO TO 250                                     00016340
16350      200 CONTINUE                                                     00016350
16360          WRITE(6,220) ITMAX                                           00016360
16370      220 FORMAT(" RELAXATION FOR THETA FAILED AFTER",I4,3X,           00016370
16380         *"ITERATIONS")                                                00016380
16390          STOP                                                         00016390
16400      250 CONTINUE                                                     00016400
16410          RETURN                                                       00016410
16420          END                                                          00016420
16430      C*-------------------------------------------------------------- 00016430
16440      C*                                                               00016440
16450          SUBROUTINE WVNUM(D,COSI,SINI,U,V,RK,A)                       00016450
16460      C*                                                               00016460
16470      C*     WVNUM CALCULATES THE WAVE NUMBER INCLUDING THE EFFECTS OF  00016470
16480      C*     WAVE-CURRENT INTERACTION                                   00016480
16490      C*                                                               00016490
16500      C*-------------------------------------------------------------- 00016500
16510          COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,     00016510
16520         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                    00016520
16530          EPSK=0.001                                                   00016530
16540          RK=PI2/(T*SQRT(G*D))                                         00016540
16550          DO 1 I=1,50                                                  00016550
16560          A=SIGMA-U*RK*COSI-RK*V*SINI                                  00016560
16570          A2=A**2                                                      00016570
16580          ARG=RK*D                                                     00016580
16590          F1=EXP(ARG)                                                  00016590
16600          F2=1.0/F1                                                    00016600
16610          SECH=2.0/(F1+F2)                                             00016610
16620          SECH2=SECH**2                                                00016620
16630          TT=TANH(ARG)                                                 00016630
16640          FK=G*RK*TT-A2                                                00016640
16650          FFK=G*(ARG*SECH2+TT)+2.0*(U*COSI+V*SINI)*A                   00016650
16660          RKNEW=RK-FK/FFK                                              00016660
16670          IF(ABS(RKNEW-RK).LE.(ABS(EPSK*RKNEW)))GO TO 3                00016670
16680          RK=RKNEW                                                     00016680
16690        1 CONTINUE                                                     00016690
16700          WRITE(6,2)I,RK,T,D,U,V                                       00016700
16710        2 FORMAT(" ITERATION FOR K FAILED TO CONVERGE: I,K,D,U,V"      00016710
16720         *,I6,5F10.5)                                                  00016720
16730          CALL EXIT                                                    00016730
16740          RETURN                                                       00016740
16750        3 RK=RKNEW                                                     00016750
16760          A=SIGMA-RK*U*COSI-RK*V*SINI                                  00016760
16770      C*-------------------------------------------------------------- 00016770
16780      C*     CHECK IF RK NEGATIVE; THIS CONDITION MAY ARRISE IF CURRENT IS 00016780
```

```
C*     TOO STRONG                                                       00016790
C*--------------------------------------------------------------------- 00016800
       IF(RK.GT.0.0)GO TO 5                                             00016810
       WRITE(6,4)D,COSI,SINI,U,V,RK,A                                   00016820
  4    FORMAT(" RK IS NEGATIVE: D,COSI,SINI,U,V,RK,A",                  00016830
      *7F10.5)                                                          00016840
       CALL EXIT                                                        00016850
  5    RETURN                                                           00016860
       END                                                              00016870
C*--------------------------------------------------------------------- 00016880
C*                                                                      00016890
       SUBROUTINE SNELL(THETA0,HH,ITER)                                 00016900
C*                                                                      00016910
C*     SNELL CALCULATES THE FIRST GUESS AT THE REFRACTION ANGLE         00016920
C*     AT EACH GRID BY SNELL'S LAW, FOR EACH ITERATION IN WHICH THE     00016930
C*     DEEPWATER WAVE HEIGHT IS BUILDING UP                             00016940
C*                                                                      00016950
C*     SNELL ALSO CALCULATES THE WAVE HEIGHT AT THE M GRID ROW          00016960
C*                                                                      00016970
C*--------------------------------------------------------------------- 00016980
       COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),  00016990
      *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)  00017000
      *,DDDY(50,50)                                                     00017010
       COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,         00017020
      *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                        00017030
       I=IDRY                                                           00017040
 30    I=I+1                                                            00017050
       IF(ITER.GT.1) I=M                                                00017060
       DO 600 J=2,N1                                                    00017070
       CALL WVNUM(D(I,J-1),-1.0,0.0,0.0,0.0,RK,A)                       00017080
       AA = RK* D(I,J-1)                                                00017090
       ANG = ARSIN(SIN(THETA0)*TANH(AA))                                00017100
       ANG = PI-ANG                                                     00017110
       ARG = 2.0*AA                                                     00017120
       SHOAL = SQRT(1.0/(TANH(AA)*(1.0+ARG/SINH(ARG))))                 00017130
       REF = SQRT(COS(THETA0)/COS(ANG))                                 00017140
       WVHT = HH*SHOAL*REF                                              00017150
       HB = 0.12*TANH(AA)*PI2/RK                                        00017160
       IF(WVHT .GT. HB) WVHT = HB                                       00017170
       IN = 1                                                           00017180
       IF(WVHT .GT. HB) IN = 0                                          00017190
       SS = SIN(ANG)                                                    00017200
       CC = COS(ANG)                                                    00017210
 43    CONTINUE                                                         00017220
       H(I,J) = WVHT                                                    00017230
       IB(I,J) = IN                                                     00017240
       Z(I,J) = ANG                                                     00017250
       SI(I,J) = SS                                                     00017260
       CO(I,J) = CC                                                     00017270
600    CONTINUE                                                         00017280
       IF(I.EQ.M) GO TO 85                                              00017290
       GO TO 30                                                         00017300
 85    CONTINUE                                                         00017310
C*--------------------------------------------------------------------- 00017320
C*     BOUNDARY CONDITIONS                                              00017330
C*--------------------------------------------------------------------- 00017340
       L = 1                                                            00017350
       K = N                                                            00017360
       J = 1                                                            00017370
 45    DO 60 I = 1,M                                                    00017380
       H(I,J) = H(I,K)                                                  00017390
       IB (I,J) = IB(I,K)                                               00017400
```

```
17410           Z(I,J)=Z(I,K)                                                   00017410
17420           SI(I,J) = SI(I,K)                                               00017420
17430        60 CO(I,J) = CO(I,K)                                               00017430
17440           L = L+1                                                         00017440
17450           IF (L .GT. 3) RETURN                                            00017450
17460           IF (L .EQ. 3)  GO TO 610                                        00017460
17470           L = 2                                                           00017470
17480           J = 2                                                           00017480
17490           K=N1                                                            00017490
17500           GO TO 45                                                        00017500
17510       610 J = N2                                                          00017510
17520           K = 3                                                           00017520
17530           GO TO 45                                                        00017530
17540           RETURN                                                          00017540
17550           END                                                             00017550
17560     C*--------------------------------------------------------------------00017560
17570     C*                                                                    00017570
17580           SUBROUTINE DGRAD                                                00017580
17590     C*                                                                    00017590
17600     C*    DGRAD CALCULATES THE SPATIAL GRADIENTS IN DEPTH AFTER EACH       00017600
17610     C*    UPDATING OF THE TOTAL DEPTH                                     00017610
17620     C*                                                                    00017620
17630     C*--------------------------------------------------------------------00017630
17640           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),  00017640
17650          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)  00017650
17660          *,DDDY(50,50)                                                    00017660
17670           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,        00017670
17680          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00017680
17690           DO 1 I = 2,M1                                                   00017690
17700           DO 1 J = 2,N1                                                   00017700
17710           DDDX(I,J) =-(D(I-1,J)-D(I+1,J))/(2.*DX)                         00017710
17720           DDDY(I,J) =-(D(I,J-1)-D(I,J+1))/(2.*DY)                         00017720
17730         1 CONTINUE                                                        00017730
17740           DO 2 I = 1,M                                                    00017740
17750           DDDX(I,1)=DDDX(I,N)                                             00017750
17760           DDDY(I,1)=DDDY(I,N)                                             00017760
17770           DDDX(I,2)=DDDX(I,N1)                                            00017770
17780           DDDY(I,2)=DDDY(I,N1)                                            00017780
17790           DDDX(I,N2)=DDDX(I,3)                                            00017790
17800           DDDY(I,N2)=DDDY(I,3)                                            00017800
17810         2 CONTINUE                                                        00017810
17820           RETURN                                                          00017820
17830           END                                                             00017830
17840     C*--------------------------------------------------------------------00017840
17850     C*                                                                    00017850
17860           SUBROUTINE TAUSB (WIND, WINANG, CF,ITO)                         00017860
17870     C*                                                                    00017870
17880     C*    THIS SUBROUTINE CALCULATES SURFACE SHEAR STRESS BY VAN DORN'S    00017880
17890     C*    METHOD, AND AN AVERAGE BOTTOM SHEAR STRESS USING A SMALL        00017890
17900     C*    CURRENT ASSUMPTION                                              00017900
17910     C*                                                                    00017910
17920     C*--------------------------------------------------------------------00017920
17930           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),  00017930
17940          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50), 00017940
17950          *DDDY(50,50),W(50,50),Y(50,50)                                   00017950
17960           COMMON/VAL/ETA(50,50)                                           00017960
17970           COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50), 00017970
17980          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                          00017980
17990           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                     00017990
18000           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,        00018000
18010          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00018010
18020     C*--------------------------------------------------------------------00018020
```

```
18030     C*     CALCULATE WIND SHEAR STRESS                                          00018030
18040     C*--------------------------------------------------------------------     00018040
18050           IF(IT.GT.ITO+1) GO TO 21                                             00018050
18060           VANCON=1.1E-06                                                       00018060
18070           IF (WIND .GE.7.2) VANCON = 1.1E-06+2.5E-06*((1.-7.2/WIND)**2.)       00018070
18080           WINDX = WIND*COS(WINANG)                                             00018080
18090           WINDY = WIND*SIN(WINANG)                                             00018090
18100           CON1  = RHO*VANCON*ABS(WIND)                                         00018100
18110           DO 20 J = 1,N2                                                       00018110
18120           DO 20 I = 2, M                                                       00018120
18130           TAUSX(I,J) =- CON1 *WINDX                                            00018130
18140           TAUSY(I,J) = CON1 *WINDY                                             00018140
18150        20 CONTINUE                                                           00018150
18160     C*--------------------------------------------------------------------     00018160
18170     C*     CALCULATE BOTTOM FRICTION AT EACH GRID POINT                        00018170
18180     C*--------------------------------------------------------------------     00018180
18190      21   DO 40 J = 2,N1                                                      00018190
18200           K = J-1                                                              00018200
18210           DO 40 I = IWET,M1                                                    00018210
18220           TAUBX(I,J) = (.5*CF*RHO*H(I,J)*W(I,J))/(T*SINH(RKA(I,J)*D(I,K)))    00018220
18230           TAUBY(I,J) = (.25*CF*RHO*H(I,J)*Y(I,J))/(T*SINH(RKA(I,J)*D(I,K)))   00018230
18240        40 CONTINUE                                                           00018240
18250           DO 100 I = 1,M                                                       00018250
18260           TAUBX(I,1) = TAUBX(I,N)                                              00018260
18270           TAUBX(I,N1) = TAUBX( I,2)                                            00018270
18280           TAUBX(I,N2) = TAUBX(I,3)                                             00018280
18290           TAUBY(I,1) = TAUBY(I,N)                                              00018290
18300           TAUBY(I,N1) = TAUBY( I,2)                                            00018300
18310           TAUBY(I,N2) = TAUBY(I,3)                                             00018310
18320        100 CONTINUE                                                          00018320
18330           RETURN                                                               00018330
18340           END                                                                  00018340
18350     C*--------------------------------------------------------------------     00018350
18360           SUBROUTINE UCALC                                                     00018360
18370     C*                                                                         00018370
18380     C*     THIS SUBROUTINE CALCULATES THE MEAN HORIZONTAL VELOCITIES           00018380
18390     C*     USING THE FINITE DIFFERENCE EQUATIONS                               00018390
18400     C*                                                                         00018400
18410     C*     NOTE THAT ALL DEPTHS USED ARE OFFSET BY -DY                         00018410
18420     C*                                                                         00018420
18430     C*--------------------------------------------------------------------     00018430
18440           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),      00018440
18450          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),     00018450
18460          *DDDY(50,50),W(50,50),Y(50,50)                                       00018460
18470           COMMON/VAL/ETA(50,50)                                                00018470
18480           COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50),   00018480
18490          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                              00018490
18500           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                          00018500
18510           COMMON/CONST/G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,              00018510
18520          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                           00018520
18530           DO 21 J=3,N1                                                         00018530
18540           K=J-1                                                                00018540
18550           DO 21 I=IWET,M-1                                                     00018550
18560           DBAR = (D(I,K)+D(I-1,K))/2.                                          00018560
18570           U(I,J) = U(I,J) +DT*((ETA(I-1,J) - ETA(I,J))*G/DX+(SIGXX(I-1,J)-SI   00018570
18580          1GXX(I,J))/(DX*RHO*DBAR ) - (SIGXY(I,J+1)-SIGXY(I,J-1)+SIGXY(I-1,J   00018580
18590          2+1)-SIGXY(I-1,J-1))/(4.*DY*RHO*DBAR)+(TAUSX(I,J)+TAUSX(I-1,J))/(2.  00018590
18600          3*RHO*DBAR)-(TAUBX(I,J)+TAUBX(I-1,J))/(2.*RHO*DBAR))                 00018600
18610           U(IWET,J)=0.0                                                        00018610
18620           DBAR = (D(I,K) + D(I,K-1))/2.0                                       00018620
18630           V(I,J)=V(I,J) +DT*((ETA(I,J-1)-ETA(I,J))*G/DY-(SIGXY(I+1,J-1)-SIGX   00018630
18640          1Y(I-1,J-1))/(4.*DX*RHO*DBAR)-(SIGXY(I+1,J)-SIGXY(I-1,J))/(4.*DX*RH  00018640
```

```
18650         20*DBAR)+(SIGYY(I,J-1) - SIGYY(I,J))/(DY*RHO*DBAR)+(TAUSY(I,J) + TA   00018650
18660         3USY(I,J-1))/(2.*RHO*DBAR) - (TAUBY(I,J) + TAUBY(I,J-1))/(2.*RHO*DB   00018660
18670         4AR))                                                                 00018670
18680        21 CONTINUE                                                            00018680
18690      C*--------------------------------------------------------------------   00018690
18700      C*     BOUNDARY CONDITIONS                                               00018700
18710      C*--------------------------------------------------------------------   00018710
18720           DO 60 I=1,M                                                        00018720
18730           V(I,N2) = V(I,3)                                                   00018730
18740           U(I,N2) = U(I,3)                                                   00018740
18750           V(I,2) = V(I,N1)                                                   00018750
18760           U(I,2) = U(I,N1)                                                   00018760
18770           V(I,1) = V(I,N)                                                    00018770
18780        60 U(I,1) = U(I,N)                                                    00018780
18790           DO 69 I=1,M1                                                       00018790
18800           DO 69 J=2,N1                                                       00018800
18810           U(M,J)=0.0                                                         00018810
18820           W(I,J)=(U(I,J)+U(I+1,J))/2.0                                       00018820
18830           Y(I,J)=(V(I,J)+V(I,J+1))/2.0                                       00018830
18840        69 CONTINUE                                                           00018840
18850           DO 79 I=1,M                                                        00018850
18860           Y(I,N2)=Y(I,3)                                                     00018860
18870           W(I,N2)=W(I,3)                                                     00018870
18880           Y(I,1)=Y(I,N)                                                      00018880
18890        79 W(I,1)=W(I,N)                                                      00018890
18900           DO 80 J=1,N+2                                                      00018900
18910        80 Y(M,J)=(V(M,J)+V(M,J+1))/2.0                                       00018910
18920           RETURN                                                             00018920
18930           END                                                                00018930
18940      C*--------------------------------------------------------------------   00018940
18950      C*                                                                       00018950
18960           SUBROUTINE ETAS                                                    00018960
18970      C*                                                                       00018970
18980      C*     ETAS CALCULATES THE CORRECTION TO SETUP AND TOTAL DEPTH BASED     00018980
18990      C*     ON CONSERVATION OF MASS                                           00018990
19000      C*                                                                       00019000
19010      C*--------------------------------------------------------------------   00019010
19020           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),    00019020
19030          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),   00019030
19040          *DDDY(50,50),W(50,50),Y(50,50)                                      00019040
19050           COMMON/VAL/ETA(50,50)                                              00019050
19060           COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50), 00019060
19070          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                             00019070
19080           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                        00019080
19090           COMMON/CONST/ G,PI,PI2,RAD,EPS,DX,DY,DT,DX2,DY2,T,SIGMA,           00019090
19100          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                          00019100
19110           DO 1 J = 3,N1                                                      00019110
19120           K = J-1                                                            00019120
19130           DO 1 I=IWET,M1                                                     00019130
19140           ETAOLD = ETA(I,J)                                                  00019140
19150           ETA(I,J)=ETAOLD+DT*(U(I,J)*(D(I,K)+D(I-1,K))/(2.*DX)     - U(I+1,J)  00019150
19160          1*(D(I,K)+D(I+1,K))/(2.*DX)+V(I,J)*(D(I,K-1)+D(I,K))/(2.*DY) - V(I,  00019160
19170          2J+1)*(D(I,K+1)+D(I,K))/(2.*DY))                                    00019170
19180           D(I,K) = D(I,K) - ETAOLD + ETA(I,J)                                00019180
19190         1 CONTINUE                                                           00019190
19200      C*--------------------------------------------------------------------   00019200
19210      C*     BOUNDARY CONDITIONS                                               00019210
19220      C*--------------------------------------------------------------------   00019220
19230           DO 2 I = 1,M                                                       00019230
19240           D(I,1)=D(I,N)                                                      00019240
19250           D(I,N2) = D(I,3)                                                   00019250
19260           D(I,N1) = D(I,2)                                                   00019260
```

```
19270           ETA(I,N2) = ETA(I,3)                                    00019270
19280           ETA(I,2) = ETA(I,N1)                                    00019280
19290       2   ETA(I,1) = ETA(I,N)                                     00019290
19300           DO 3 I=1,M                                              00019300
19310           IF(D(I,1).GT.DD) GO TO 4                                00019310
19320       3   CONTINUE                                                00019320
19330       4   IWET=I                                                  00019330
19340           IDRY=IWET-1                                             00019340
19350           RETURN                                                  00019350
19360           END                                                     00019360
```

NONLINEAR MODEL

```
1000    $RESET FREE                                                              00001000
1100    C*-------------------------------------------------------------------   00001100
1200    C*                                                                       00001200
1300    C*    NEARSHORE CIRCULATION MODEL --- NONLINEAR VERSION                  00001300
1400    C*                                                                       00001400
1500    C*    THIS PROGRAM IS A NUMERICAL MODEL TO PREDICT SURF ZONE DYNAMICS,    00001500
1600    C*    GIVEN A PERIODIC DEPTH GRID AND MONOCHROMATIC WAVE FIELD AS INPUT.  00001600
1700    C*    THE REFRACTION PROGRAM, INCLUDING THE EFFECTS OF WAVE-CURRENT      00001700
1800    C*    INTERACTION, WAS DEVELOPED BY NODA ET. AL. (1974).  THE LINEAR     00001800
1900    C*    VERSION OF THIS PROGRAM WAS DEVELOPED BY BIRKEMEIER AND DALRYMPLE  00001900
2000    C*    (1977).  THE PRESENT NONLINEAR VERSION UTILISES THE FULL NON-      00002000
2100    C*    LINEAR MOMENTUM EQUATIONS (EBERSOLE AND DALRYMPLE(1979).  BOTTOM   00002100
2200    C*    FRICTION IS CALCULATED USING THE METHOD DEVELOPED BY DALRYMPLE AND 00002200
2300    C*    LIU                                                                00002300
2400    C*                                                                       00002400
2500    C*-------------------------------------------------------------------   00002500
2600          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),    00002600
2700         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),   00002700
2800         *DDDY(50,50),W(50,50),Y(50,50)                                      00002800
2900          COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),            00002900
3000         *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)        00003000
3100          COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50), 00003100
3200         *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                             00003200
3300          COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                        00003300
3400          COMMON/EDDY/EX(50,50),EY,VCON                                      00003400
3500          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,     00003500
3600         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                          00003600
3700          DIMENSION DST(50,50)                                               00003700
3800    C*-------------------------------------------------------------------   00003800
3900    C*                                                                       00003900
4000    C*    NOTES ON RUNNING THE PROGRAM                                       00004000
4100    C*                                                                       00004100
4200    C*    1. WAVE ANGLE IS MEASURED CLOCKWISE FROM THE +X DIRECTION          00004200
4300    C*                                                                       00004300
4400    C*    2. WIND ANGLE IS MEASURED CLOCKWISE FROM THE -X DIRECTION          00004400
4500    C*                                                                       00004500
4600    C*    3. ALL INPUT AND OUTPUT VARIABLES ARE IN MKS UNITS                 00004600
4700    C*                                                                       00004700
4800    C*    4. THE DEPTH GRID IS OFFSET -DY FROM ALL OTHER VARIABLE GRIDS      00004800
4900    C*                                                                       00004900
5000    C*-------------------------------------------------------------------   00005000
5100    C*                                                                       00005100
5200    C*    DEFINITIONS OF QUANTITIES USED IN PROGRAM                          00005200
5300    C*                                                                       00005300
5400    C*    1. CONSTANTS                                                       00005400
5500    C*                                                                       00005500
5600    C*       CF - BOTTOM FRICTION COEFFICIENT (DARCY TYPE)                   00005600
5700    C*       VANCON - WIND SHEAR COEFFICIENT (VAN DORN TYPE)                 00005700
5800    C*       EY - LATERAL EDDY COEFFICIENT, LONGSHORE DIRECTION              00005800
5900    C*       VCON - EDDY VISCOSITY COEFFICIENT, OFFSHORE DIRECTION           00005900
6000    C*                                                                       00006000
6100    C*    2. VARIABLE ARRAYS (VALUE AT EACH GRID LOCATION)                   00006100
6200    C*                                                                       00006200
6300    C*       D - TOTAL WATER DEPTH, STILL WATER + SETUP(ETA)                  00006300
6400    C*       DDDX,DDDY - LOCAL DERIVATIVES OF THE TOTAL DEPTH                00006400
6500    C*       ETA - DEVIATION OF MEAN WATER LEVEL FROM STILL WATER LEVEL       00006500
6600    C*       Z - WAVE ANGLE                                                  00006600
6700    C*       CO - COS(Z)                                                     00006700
6800    C*       SI - SIN(Z)                                                     00006800
```

```
C*       H - WAVE HEIGHT                                                    00006900
C*       HBREAK - LOCAL BREAKING WAVE HEIGHT                                00007000
C*       IB - BREAKING INDEX                                                00007100
C*            IB=0, WAVE IS BREAKING LOCALLY                                00007200
C*            IB=1, WAVE IS NOT BREAKING LOCALLY                            00007300
C*       CG - GROUP VELOCITY                                                00007400
C*       RKA - WAVE NUMBER                                                  00007500
C*       U,V - X,Y VELOCITIES AT SIDES OF GRID BLOCKS                       00007600
C*       W,Y - X,Y VELOCITIES AT CENTERS OF GRID BLOCKS                     00007700
C*             NOTE THAT THE U,V AND W,Y ARRAYS ARE INTERCHANGED            00007800
C*             IN SEVERAL SUBROUTINES THROUGH THE COMMON STATEMENTS         00007900
C*       SIGX,SIGXY,SIGY - RADIATION STRESSES                               00008000
C*       TAUBX,TAUBY - BOTTOM STRESSES                                      00008100
C*       TAUSX,TAUSY - SURFACE STRESSES                                     00008200
C*       EX - LATERAL EDDY COEFFICIENT, OFFSHORE DIRECTION                  00008300
C*                                                                          00008400
C*    3. LOCALLY DEFINED VARIABLES                                          00008500
C*                                                                          00008600
C*       DUDX,DVDX - VELOCITY GRADIENT COMPONANTS                           00008700
C*       DCDX,DCDY - WAVE CELERITY GRADIENT COMPONANTS                      00008800
C*       DCGDX,DCGDY - GROUP VELOCITY GRADIENT COMPONANTS                   00008900
C*       DTDX,DTDY - WAVE ANGLE GRADIENT COMPONANTS                         00009000
C*       EPS - ACCURACY VALUE USED IN RELAXATION SCHEMES                    00009100
C*                                                                          00009200
C*----------------------------------------------------------------          00009300
C*                                                                          00009400
C*    VARIABLES TO BE READ INTO PROGRAM                                     00009500
C*                                                                          00009600
C*    1. WAVE PARAMETERS (DEEPWATER)                                        00009700
C*                                                                          00009800
C*       T        - WAVE PERIOD (SECONDS)                                  00009900
C*       HO       - WAVE HEIGHT (METERS)                                   00010000
C*       A        - WAVE ANGLE, CLOCKWISE FROM +X (DEGREES)                00010100
C*                                                                          00010200
C*    2. WIND PARAMETERS                                                    00010300
C*                                                                          00010400
C*       WIND     - WIN SPEED (METERS/SECOND)                              00010500
C*       WINANG   - WIND ANGLE, CLOCKWISE FROM -X (DEGREES)                00010600
C*                                                                          00010700
C*    3. GRID PARAMETERS                                                    00010800
C*                                                                          00010900
C*       M        - NUMBER OF GRIDS IN X DIRECTION                         00011000
C*       N        - NUMBER OF GRIDS IN Y DIRECTION                         00011100
C*       DX       - GRID SPACING IN X DIRECTION (METERS)                   00011200
C*       DY       - GRID SPACING IN Y DIRECTION (METERS)                   00011300
C*       DT       - TIME STEP, <DX/SQRT(2*G*DMAX) (SECONDS)                00011400
C*       INDEX    - SPECIFY INPUT OF DEPTH INFORMATION                     00011500
C*                  =1, READ DATA FROM PREVIOUS RUN                        00011600
C*                  =2, READ DEPTH GRID FROM INPUT DATA FILE               00011700
C*                  =3, ESTABLISH PLANE BEACH BASED ON INPUT BEACH SLOPE   00011800
C*       AM       - BEACH SLOPE                                            00011900
C*                                                                          00012000
C*    4. PROGRAM CONTROL PARAMETERS                                         00012100
C*                                                                          00012200
C*       ITA      - TOTAL NUMBER OF ITERATIONS (INCLUDING ACCUMULATED      00012300
C*                  TOTAL OF PREVIOUS RUNS IF INDEX=1 IS USED)             00012400
C*       NHIGHT   - NUMBER OF ITERATIONS OVER WHICH THE DEEP WATER WAVE     00012500
C*                  HEIGHT IS BUILT UP                                     00012600
C*       ID       - DETERMINE IF DISSAPATION OF WAVE ENERGY IS TO BE       00012700
C*                  INCLUDED                                               00012800
C*                  =0, NO DISSAPATION                                     00012900
C*                  =1, DISSAPATION DUE TO VISCOSITY AND BOTTOM PERM-      00013000
```

```
13100    C*                      -EABILITY                                      00013100
13200    C*       KSKIP   - DETERMINE FREQUENCY OF PRINTED ITERATIONS           00013200
13300    C*                                                                     00013300
13400    C*    5. DEPTH GRID (IF INDEX=2)                                       00013400
13500    C*                                                                     00013500
13600    C*       D       - LOCAL WATER DEPTH W/R MEAN WATER LEVEL.             00013600
13700    C*                 SEE USERS MANUAL FOR INPUT FORMAT                   00013700
13800    C*                                                                     00013800
13900    C*-------------------------------------------------------------       00013900
14000          RHO=1000.                                                       00014000
14100          G=9.81                                                          00014100
14200          PI=3.1415927                                                    00014200
14300          PI2=PI *2.0                                                     00014300
14400          RAD=180.0/PI                                                    00014400
14410          CF=0.010                                                        00014410
14420          DD=0.0                                                          00014420
14430          EY=0.25                                                         00014430
14440          VCON=0.0025                                                     00014440
14500    C*-------------------------------------------------------------------00014500
14600    C*    READ INPUT DATA                                                  00014600
14700    C*                                                                     00014700
14800    C*    1. WAVE PARAMETERS                                               00014800
14900          READ(5,/)T,HO,A                                                 00014900
15000    C*    2. WIND PARAMETERS                                               00015000
15100          READ(5,/)WIND,WINANG                                            00015100
15200    C*    3. GRID PARAMETERS                                               00015200
15300          READ(5,/)M,N,DX,DY,DT,INDEX,AM                                  00015300
15400    C*    4. PROGRAM CONTROL PARAMETERS                                    00015400
15500          READ(5,/)ITA,NHIGHT,ID,KSKIP                                    00015500
15600    C*-------------------------------------------------------------------00015600
15700    C*    WRITE INPUT DATA ONTO OUTPUT FILE 6 AS HEADER                    00015700
15800    C*-------------------------------------------------------------------00015800
15900          WRITE(6,104)M,N,ITA                                             00015900
16000          WRITE(6,102)DX,DY,DT                                            00016000
16100          WRITE(6,105) T,HO,A,AM                                          00016100
16200          WRITE(6,106) WIND,WINANG                                        00016200
16300          WRITE(6,119) IDDEX,IN,MM                                        00016300
16400          WRITE(6,121) NHIGHT                                             00016400
16500          WRITE(6,120) CF,DD                                              00016500
16600          WRITE(6,123) EY,VCON                                            00016600
16700          ITO=0                                                           00016700
16800          HGT=HO                                                          00016800
16900          DELTAT=DT                                                       00016900
17000          WINANG=WINANG/RAD                                               00017000
17100          N2=N+2                                                          00017100
17200          N1=N+1                                                          00017200
17300          M2=M-2                                                          00017300
17400          M1=M-1                                                          00017400
17500          DX2=DX*2.                                                       00017500
17600          DY2=DY*2.                                                       00017600
17700          SIGMA=PI2/T                                                     00017700
17800          EPSH=0.001                                                      00017800
17900          EPSA=0.01                                                       00017900
18000    C*-------------------------------------------------------------       00018000
18100    C*    ESTABLISH BOTTOM TOPOGRAPHY, BASED ON VALUE OF "INDEX"           00018100
18200    C*-------------------------------------------------------------       00018200
18300          GO TO (1,2,3) INDEX                                             00018300
18400    C*-------------------------------------------------------------------00018400
18500    C*    INDEX=1, READ DATA FROM FIE 8 FOR PREVIOUS RUN OF MODEL          00018500
18600    C*-------------------------------------------------------------------00018600
18700       1  READ(8,116)                                                     00018700
18800         *((ETA(I,J),J=1,N2),I=1,M),((D(I,J),J=1,N2),I=1,M),             00018800
```

```
18900          *((U(I,J),J=1,N2),I=1,M),((V(I,J),J=1,N2),I=1,M),                00018900
19000          *((UM1(I,J),J=1,N2),I=1,M),((VM1(I,J),J=1,N2),I=1,M),            00019000
19100          *((DM1(I,J),J=1,N2),I=1,M),((ETAM1(I,J),J=1,N2),I=1,M),          00019100
19200          *((Z(I,J),J=1,N2),I=1,M),((H(I,J),J=1,N2),I=1,M),                00019200
19300          *((W(I,J),J=1,N2),I=1,M),((Y(I,J),J=1,N2),I=1,M)                 00019300
19400          *,((RKA(I,J),J=1,N2),I=1,M)                                      00019400
19500           READ(8,118)((SIGXX(I,J),J=1,N2),I=1,M),                         00019500
19600          *((SIGXY(I,J),J=1,N2),I=1,M),((SIGYY(I,J),J=1,N2),I=1,M)         00019600
19700           READ(8,117)ITQ                                                  00019700
19800           DO 450 I=1,M                                                    00019800
19900           DO 450 J=1,N2                                                   00019900
20000           CO(I,J)=COS(Z(I,J))                                             00020000
20100           SI(I,J)=SIN(Z(I,J))                                             00020100
20200       450 CONTINUE                                                        00020200
20300     C*--------------------------------------------------------------------00020300
20400     C*     IF WAVE ANGLE IS LESS THAN 180 DEGREES, VARIABLE GRIDS WERE   00020400
20500     C*     FLIPPED IN PREVIOUS RUN.  ROTATE WAVE AND WIND ANGLES         00020500
20600     C*--------------------------------------------------------------------00020600
20700           IF(A.GE.180.) GO TO 31                                          00020700
20800           A=360.-A                                                        00020800
20900           WINANG=PI2-WINANG                                               00020900
21000           GO TO 31                                                        00021000
21100     C*--------------------------------------------------------------------00021100
21200     C*     INDEX=2, READ DEPTH GRID FROM INPUT FILE 5                    00021200
21300     C*--------------------------------------------------------------------00021300
21400       2   READ (5,/) ((D(I,J),J=1,N),I=1,M)                               00021400
21500     C*--------------------------------------------------------------------00021500
21600     C*     IF WAVE ANGLE IS LESS THAN 180 DEGREES, FLIP THE INPUT DEPTH  00021600
21700     C*     GRID AND ROTATE THE WAVE AND WIND ANGLES                      00021700
21800     C*--------------------------------------------------------------------00021800
21900           IF(A.GT.180.) GO TO 161                                         00021900
22000           A=360.-A                                                        00022000
22100           WINANG=PI2-WINANG                                               00022100
22200           DO 158 I=1,M                                                    00022200
22300           DO 159 J=1,N                                                    00022300
22400       159 DST(I,J)=D(I,J)                                                 00022400
22500           DO 158 J=1,N                                                    00022500
22600           K=(N-J)+1                                                       00022600
22700       158 D(I,K)=DST(I,J)                                                 00022700
22800       161 DO 160 I=1,M                                                    00022800
22900           D(I,1)=D(I,N)                                                   00022900
23000           D(I,N2)=D(I,3)                                                  00023000
23100           D(I,N1)=D(I,2)                                                  00023100
23200       160 CONTINUE                                                        00023200
23300           GO TO 146                                                       00023300
23400     C*--------------------------------------------------------------------00023400
23500     C*     INDEX=3, ESTABLISH PLANE BEACH                                00023500
23600     C*--------------------------------------------------------------------00023600
23700       3   DO 30 J=1,N2                                                    00023700
23800           DO 30 I=1,M                                                     00023800
23900           D(I,J)=((FLOAT(I-2)*DX)*AM)                                     00023900
24000       30  CONTINUE                                                       00024000
24100     C*--------------------------------------------------------------------00024100
24200     C*     IF WAVE ANGLE IS LESS THAN 180 DEGREES, ROTATE WAVE AND WIND ANGLE 00024200
24300     C*--------------------------------------------------------------------00024300
24400           IF(A.GE.180.) GO TO 146                                         00024400
24500           A=360.-A                                                        00024500
24600           WINANG=PI2-WINANG                                               00024600
24700     C*--------------------------------------------------------------------00024700
24800     C*     WRITE DEPTH GRID ON OUTPUT IF INDEX=2 OR 3                    00024800
24900     C*--------------------------------------------------------------------00024900
25000       146 WRITE(6,108)                                                    00025000
```

```
25100           WRITE(6,103)((D(I,J),J=1,N1),I=1,M)                                  00025100
25200     C*--------------------------------------------------------------------     00025200
25300     C*     INITIALIZE ARRAYS                                                   00025300
25400     C*--------------------------------------------------------------------     00025400
25500           DO 65 I=1,M                                                          00025500
25600           DO 65 J=1,N2                                                         00025600
25700           ETA(I,J)=0.0                                                         00025700
25800           U(I,J)=0.0                                                           00025800
25900           V(I,J)=0.0                                                           00025900
26000           ETAM1(I,J)=0.0                                                       00026000
26100           DM1(I,J)=D(I,J)                                                      00026100
26200           UM1(I,J)=0.0                                                         00026200
26300           VM1(I,J)=0.0                                                         00026300
26400        65 CONTINUE                                                           00026400
26500        31 DO 15 I=1,M                                                        00026500
26600           IF(D(I,1).GT.DD) GO TO 95                                          00026600
26700        15 CONTINUE                                                           00026700
26800        95 IWET=I                                                             00026800
26900           IDRY=IWET-1                                                        00026900
27000     C*--------------------------------------------------------------------     00027000
27100     C*     MAIN PROGRAM LOOP FOR EACH ITERATION                                00027100
27200     C*--------------------------------------------------------------------     00027200
27300           IT=ITO                                                             00027300
27400           IF(ITO.NE.0) GO TO 4                                               00027400
27500           HO=HGT*TANH(1.0/FLOAT(NHIGHT))                                     00027500
27600           CALL DGRAD                                                         00027600
27700           CALL REFRAC(A,HO,1,NHIGHT,CF)                                      00027700
27800           CALL TAUSB(WIND,WINANG,CF,ITO)                                     00027800
27900           TSTEP=DT*0.5                                                       00027900
28000           CALL CONTIN                                                        00028000
28100           CALL ETAS(ETA,TSTEP)                                               00028100
28200           CALL MOMEN                                                         00028200
28300           CALL UCALC(D,U,V,TSTEP)                                            00028300
28400           CALL ROLBAC                                                        00028400
28500        4  DO 5 IT=(1+ITO),ITA                                               00028500
28600           L=AMOD(FLOAT(IT),FLOAT(KSKIP))                                     00028600
28700           MCOUNT=AMOD(FLOAT(IT),10.0)                                        00028700
28800           HO=HGT*TANH(FLOAT(IT)/(FLOAT(NHIGHT)/2.0))                         00028800
28900     C     IF(IT.LE.((NHIGHT*2.0)+5)) GO TO 12                                00028900
29000     C     GO TO 13                                                           00029000
29100        12 CALL DGRAD                                                         00029100
29200           ITP1=IT+1                                                          00029200
29300           CALL REFRAC(A,HO,ITP1,NHIGHT,CF)                                   00029300
29400        13 CALL TAUSB(WIND,WINANG,CF,ITO)                                     00029400
29500           IF(IT.EQ.1) GO TO 51                                               00029500
29600           GO TO 52                                                           00029600
29700        51 TSTEP=DT                                                           00029700
29800           CALL CONTIN                                                        00029800
29900           CALL ETAS(ETAM1,TSTEP)                                             00029900
30000           CALL MOMEN                                                         00030000
30100           CALL UCALC(DM1,UM1,VM1,TSTEP)                                      00030100
30200           DO 66 I=1,M                                                        00030200
30300           DO 66 J=1,N2                                                       00030300
30400           ETA(I,J)=ETAP1(I,J)                                                00030400
30500           D(I,J)=DP1(I,J)                                                    00030500
30600           U(I,J)=UP1(I,J)                                                    00030600
30700           V(I,J)=VP1(I,J)                                                    00030700
30800        66 CONTINUE                                                           00030800
30900           GO TO 18                                                           00030900
31000        52 TSTEP=DT*2.0                                                       00031000
31100           CALL CONTIN                                                        00031100
31200           CALL ETAS(ETAM1,TSTEP)                                             00031200
```

```
31300          CALL MOMEN                                                    00031300
31400          CALL UCALC(DM1,UM1,VM1,TSTEP)                                 00031400
31500          IF(MCOUNT.EQ.0) GO TO 53                                      00031500
31600          CALL ROLBAC                                                   00031600
31700          GO TO 18                                                      00031700
31800       53 CALL ROLBAC                                                   00031800
31900          TSTEP=DT                                                      00031900
32000          CALL CONTIN                                                   00032000
32100          CALL ETAS(ETAM1,TSTEP)                                        00032100
32200          CALL MOMEN                                                    00032200
32300          CALL UCALC(DM1,UM1,VM1,TSTEP)                                 00032300
32400          DO 57 I=1,M                                                   00032400
32500          DO 57 J=1,N2                                                  00032500
32600          ETA(I,J)=ETAP1(I,J)                                           00032600
32700          D(I,J)=DP1(I,J)                                               00032700
32800          U(I,J)=UP1(I,J)                                               00032800
32900          V(I,J)=VP1(I,J)                                               00032900
33000       57 CONTINUE                                                      00033000
33100       18 CONTINUE                                                      00033100
33400          SUM=0.0                                                       00033400
33500          DO 86 I=1,M                                                   00033500
33600          DO 86 J=2,N                                                   00033600
33700          SUM=SUM+0.5*(W(I,J)**2+Y(I,J)**2)*D(I,J-1)                    00033700
33800          SUM=SUM+G*ETA(I,J)*(D(I,J-1)-0.5*ETA(I,J))                    00033800
33900       86 CONTINUE                                                      00033900
34000          WRITE(6,101) SUM                                              00034000
34100          IF(L.NE.0)GO TO 5                                             00034100
34200          WRITE(6,112)                                                  00034200
34300          WRITE(6,113)IT,H0                                             00034300
34400          WRITE(6,114)                                                  00034400
34500          WRITE(6,115)                                                  00034500
34600          WRITE(6,103)((H(I,J),J=1,N1),I=1,M)                           00034600
34700          DO 401 I=1,M                                                  00034700
34800          DO 401 J=1,N2                                                 00034800
34900      401 ZZ(I,J)=360.0-Z(I,J)*RAD                                      00034900
35000          WRITE(6,122)                                                  00035000
35100          WRITE(6,109)((ZZ(I,J),J=1,N1),I=1,M)                          00035100
35200          WRITE (6,107)                                                 00035200
35300          WRITE (6,103) ((U(I,J), J = 1,N1),I=1,M)                      00035300
35400          WRITE (6,110)                                                 00035400
35500          WRITE (6,103) ((V(I,J), J = 1,N1),I=1,M)                      00035500
35600          WRITE (6,111)                                                 00035600
35700          WRITE(6,103) ((ETA(I,J),J=1,N1),I=1,M)                        00035700
35800        5 CONTINUE                                                      00035800
35900     C*-------------------------------------------------------------    00035900
36000     C      THE FOLLOWING WRITE STATEMENTS WRITE OUT THE RESULTS OF     00036000
36100     C      THE LAST ITERATION ONTO AN OUTPUT FILE TO BE USED FOR A     00036100
36200     C      SUBSEQUENT START-UP                                         00036200
36300     C*-------------------------------------------------------------    00036300
36400          WRITE(3,116)                                                  00036400
36500         *((ETA(I,J),J=1,N2),I=1,M),((D(I,J),J=1,N2),I=1,M),           00036500
36600         *((U(I,J),J=1,N2),I=1,M),((V(I,J),J=1,N2),I=1,M),             00036600
36700         *((UM1(I,J),J=1,N2),I=1,M),((VM1(I,J),J=1,N2),I=1,M),         00036700
36800         *((DM1(I,J),J=1,N2),I=1,M),((ETAM1(I,J),J=1,N2),I=1,M),       00036800
36900         *((Z(I,J),J=1,N2),I=1,M),((H(I,J),J=1,N2),I=1,M),             00036900
37000         *((W(I,J),J=1,N2),I=1,M),((Y(I,J),J=1,N2),I=1,M)              00037000
37100         *,((RKA(I,J),J=1,N2),I=1,M)                                   00037100
37200          WRITE(3,118)((SIGXX(I,J),J=1,N2),I=1,M),                     00037200
37300         *((SIGXY(I,J),J=1,N2),I=1,M),((SIGYY(I,J),J=1,N2),I=1,M)      00037300
37400          WRITE(3,117)ITA                                               00037400
37500          LOCK 3                                                        00037500
37600      101 FORMAT(1X,"THE TOTAL ENERGY IS",F14.2)                        00037600
```

```
37700      102 FORMAT (/,1X,'DX=',F4.1,3X,'DY=',F4.1,3X,'DT=',F5.2)                  00037700
37800      103 FORMAT(11F10.4)                                                       00037800
37900      104 FORMAT(1X,"M=",I5,3X,"N=",I5,3X,"ITERATIONS =",I5)                   00037900
38000      105 FORMAT(1X,'WAVE PARAMETERS',/2X,'PERIOD =',F7.2,'  HEIGHT =',F7      00038000
38100         *.2,3X,7HANGLE =,F7.2,1X,7HDEGREES,4X,12HBEACH SLOPE=,F7.4)           00038100
38200      106 FORMAT (1X,'WIND SPEED =',F7.3,'M/SEC',5X,'WIND ANGLE =',F7.         00038200
38300         *3,' DEGREES')                                                        00038300
38400      108 FORMAT (//' THE AUGMENTED MATRIX OF WATER DEPTH IN METERS'/)         00038400
38500      109 FORMAT(15F8.3)                                                       00038500
38600      107 FORMAT ( 10X,12HX - VELOCITY)                                        00038600
38700      110 FORMAT ( 10X,12HY - VELOCITY)                                        00038700
38800      111 FORMAT (10X,10HETA VALUES)                                           00038800
38900      112 FORMAT('*****************************',/)                           00038900
39000      113 FORMAT(10X,'ITERATION NUMBER',I6,10X,'WAVE HEIGHT=',F10.5,/)         00039000
39100      114 FORMAT (1X,'*****************************',/)                        00039100
39200      115 FORMAT (5X,'WAVE HEIGHTS',/)                                         00039200
39300      116 FORMAT(15F8.4)                                                       00039300
39400      117 FORMAT(I5)                                                           00039400
39500      118 FORMAT(10F12.2)                                                      00039500
39600      119 FORMAT(5X,'IDDEX =',I5,3X,'IN =',I5,3X,'MM='                         00039600
39700         *,I5)                                                                 00039700
39800      120 FORMAT(10X,4HCF =,F10.3,5X,4HDD =,F10.3)                             00039800
39900      121 FORMAT(10X,'WAVE HEIGHT BUILDS UP FIRST',I3,' ITERATIONS')           00039900
40000      122 FORMAT(2X,'WAVE ANGLE (DEGREES)')                                    00040000
40100      123 FORMAT(10X,'EY =',F3.1,',  VCON= ',F6.4)                             00040100
40200          STOP                                                                 00040200
40300          END                                                                  00040300
40400    C*-------------------------------------------------------------------     00040400
40500    C*                                                                        00040500
40600          SUBROUTINE REFRAC(THETAO,HH,ITER,NHIGHT,CF)                         00040600
40700    C*                                                                        00040700
40800    C*    REFRAC CONTROLS THE CALCULATION OF WAVE HEIGHT AND DIRECTION        00040800
40900    C*                                                                        00040900
41000    C*-------------------------------------------------------------------     00041000
41100          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),     00041100
41200         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)     00041200
41300         *,DDDY(50,50)                                                        00041300
41400          COMMON/STRESS/SIGX(50,50),SIGY(50,50),SIGXY(50,50),TAUBX(50,50),    00041400
41500         *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                              00041500
41600          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,      00041600
41700         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                           00041700
41800          IF (ITER.GT.NHIGHT) GO TO 1                                         00041800
41900    C*-------------------------------------------------------------------     00041900
42000    C*    CALL SNELL DURING THE BUILDUP OF THE DEEPWATER WAVE                 00042000
42100    C*-------------------------------------------------------------------     00042100
42200          THETA=PI2-THETAO/RAD                                                00042200
42300          CALL SNELL (THETA,HH,ITER)                                          00042300
42400        1 DO 2 I=1,IDRY                                                       00042400
42500          DO 2 J=1,N2                                                         00042500
42600          H(I,J)=0.0                                                          00042600
42700          Z(I,J)=PI                                                           00042700
42800          SI(I,J)=0.0                                                         00042800
42900          CO(I,J)=-1.0                                                        00042900
43000        2 CONTINUE                                                            00043000
43100    C*-------------------------------------------------------------------     00043100
43200    C*    CALCULATE WAVE ANGLE AND WAVE HEIGHT AT EACH GRID                   00043200
43300    C*-------------------------------------------------------------------     00043300
43400          CALL ANGLE(25)                                                      00043400
43500          CALL HEIGHT(25,CF)                                                  00043500
43600    C*-------------------------------------------------------------------     00043600
43700    C*    CALCULATE THE RADIATION STRESSES                                    00043700
43800    C*-------------------------------------------------------------------     00043800
```

```
43900           DO 4 I=IWET,M1                                                  00043900
44000           DO 3 J=3,N1                                                     00044000
44100           ENERGY=0.125*RHO*G*(H(I,J)**2)                                  00044100
44200           SIGX(I,J)=SIGX(I,J)*ENERGY                                      00044200
44300           SIGY(I,J)=SIGY(I,J)*ENERGY                                      00044300
44400           SIGXY(I,J)=SIGXY(I,J)*ENERGY                                    00044400
44500       3   CONTINUE                                                        00044500
44600           SIGX(I,1)=SIGX(I,N)                                             00044600
44700           SIGY(I,1)=SIGY(I,N)                                             00044700
44800           SIGXY(I,1)=SIGXY(I,N)                                           00044800
44900           SIGX(I,2)=SIGX(I,N1)                                            00044900
45000           SIGY(I,2)=SIGY(I,N1)                                            00045000
45100           SIGXY(I,2)=SIGXY(I,N1)                                          00045100
45200           SIGX(I,N2)=SIGX(I,3)                                            00045200
45300           SIGY(I,N2)=SIGY(I,3)                                            00045300
45400           SIGXY(I,N2)=SIGXY(I,3)                                          00045400
45500       4   CONTINUE                                                        00045500
45600           DO 5 J=3,N1                                                     00045600
45700           SIGXY(M,J)=2.0*SIGXY(M1,J)-SIGXY(M2,J)                          00045700
45800       5   CONTINUE                                                        00045800
45900           SIGXY(M,1)=SIGXY(M,N)                                           00045900
46000           SIGXY(M,2)=SIGXY(M,N1)                                          00046000
46100           SIGXY(M,N2)=SIGXY(M,3)                                          00046100
46200           RETURN                                                          00046200
46300           END                                                             00046300
46400       C*-------------------------------------------------------------------00046400
46500       C*                                                                  00046500
46600           SUBROUTINE GROUP(I,J,DCGDX,DCGDY,FF)                            00046600
46700       C*                                                                  00046700
46800       C*     THIS SUBROUTINE CALCULATES THE GROUP VELOCITY ,WAVE CELERITY  00046800
46900       C*     AND THEIR SPATIAL DERIVATIVES.                               00046900
47000       C*                                                                  00047000
47100       C*-------------------------------------------------------------------00047100
47200           COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),  00047200
47300          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00047300
47400          *,DDDY(50,50),U(50,50),V(50,50)                                  00047400
47500           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00047500
47600          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00047600
47700           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                     00047700
47800           DUDX(I,J)=(W(I+1,J)-W(I,J))/DX                                  00047800
47900           DUDY(I,J)=(U(I,J+1)-U(I,J-1))/DY2                               00047900
48000           DVDX(I,J)=(V(I+1,J)-V(I-1,J))/DX2                               00048000
48100           DVDY(I,J)=(Y(I,J+1)-Y(I,J))/DY                                  00048100
48200           DTDX(I,J)=(Z(I+1,J)-Z(I-1,J))/DX2                               00048200
48300           DTDY(I,J)=(Z(I,J+1)-Z(I,J-1))/DY2                               00048300
48400           E(I,J)=U(I,J)*COSI+V(I,J)*SINI+0.5*A*(1.0+ARG2/SINH2)/RK        00048400
48500           DKDX(I,J)=RK*((U(I,J)*SINI-V(I,J)*COSI)*DTDX(I,J)-(COSI*DUDX(I,J)+ 00048500
48600          *SINI*DVDX(I,J))-A*DDDX(I,J-1)/SINH2)/EE                         00048600
48700           DKDY(I,J)=RK*((U(I,J)*SINI-V(I,J)*COSI)*DTDY(I,J)-(COSI*DUDY(I,J) 00048700
48800          * +SINI*DVDY(I,J))-A*DDDY(I,J-1)/SINH2)/EE                       00048800
48900           JJ=J-1                                                          00048900
49000           DEP=D(I,JJ)                                                     00049000
49100           COSI=CO(I,J)                                                    00049100
49200           SINI=SI(I,J)                                                    00049200
49300           DCGDX=0.0                                                       00049300
49400           DCGDY=0.0                                                       00049400
49500       C*  CHECK FOR DRY LAND                                             00049500
49600           IF(DEP.LE.DD) RETURN                                            00049600
49700           CALL WVNUM(DEP,COSI,SINI,U(I,J),V(I,J),RK,A)                    00049700
49800           RKA(I,J) = RK                                                   00049800
49900           IF (J.EQ.N) RKA(I,1)=RKA(I,J)                                   00049900
50000           IF (J.EQ.2) RKA(I,N1)=RKA(I,J)                                  00050000
```

```
50100           IF (J.EQ.3) RKA(I,N2)=RKA(I,J)                                  00050100
50200           TA=TANH(RK*DEP)                                                 00050200
50300           HBREAK(I,J)=0.12*PI2*TA/RK                                      00050300
50400           COSH1=COSH(RK*DEP)                                              00050400
50500           SECHSQ=1.0/(COSH1**2)                                           00050500
50600           ARG2=2.0*RK*DEP                                                 00050600
50700           SINH2=SINH(ARG2)                                                00050700
50800           COSH2=COSH(ARG2)                                                00050800
50900           SINHSQ=SINH2**2                                                 00050900
51000           EE=E(I,J)                                                       00051000
51100           C=SQRT(G*TA/RK)                                                 00051100
51200           FF=0.5*(1.0+ARG2/SINH2)                                         00051200
51300           CG(I,J)=FF*C                                                    00051300
51400           P=C*(SINH2-ARG2*COSH2)/SINHSQ                                   00051400
51500           DKDDX=RK*DDDX(I,JJ)+DEP*DKDX(I,J)                               00051500
51600           DKDDY=RK*DDDY(I,JJ)+DEP*DKDY(I,J)                               00051600
51700           Q=0.5*G/(C*RK**2)                                               00051700
51800           DCDX=Q*(RK*SECHSQ*DKDDX-TA*DKDX(I,J))                           00051800
51900           DCDY=Q*(RK*SECHSQ*DKDDY-TA*DKDY(I,J))                           00051900
52000           DCGDX=P*DKDDX+FF*DCDX                                           00052000
52100           DCGDY=P*DKDDY+FF*DCDY                                           00052100
52200           RETURN                                                          00052200
52300           END                                                             00052300
52400     C*--------------------------------------------------------------------00052400
52500     C*                                                                    00052500
52600           SUBROUTINE HEIGHT(ITMAX,CF)                                     00052600
52700     C                                                                     00052700
52800     C*    HEIGHT CALCULATES THE LOCAL WAVE HEIGHT AT EACH GRID LOCATION   00052800
52900     C*                                                                    00052900
53000     C*--------------------------------------------------------------------00053000
53100           COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),  00053100
53200          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00053200
53300          *,DDDY(50,50),U(50,50),V(50,50)                                  00053300
53400           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00053400
53500          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00053500
53600           COMMON/STRESS/SIGX(50,50),SIGY(50,50),SIGXY(50,50),TAUBX(50,50), 00053600
53700          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                          00053700
53800           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                     00053800
53900            VISCOS=1.0E-05                                                 00053900
54000            PERM=1.06E-09                                                  00054000
54200           DO 500 I=2,M1                                                   00054200
54300           DO 500 J=2,N1                                                   00054300
54400           IF(D(I,J-1).LE.DD) GO TO 499                                    00054400
54500           CALL GROUP(I,J,DCGDX,DCGDY,FF)                                  00054500
54600           DUDX=(W(I+1,J)-W(I,J))/DX                                       00054600
54700           DUDY=(U(I,J+1)-U(I,J-1))/DY2                                    00054700
54800           DVDX=(V(I+1,J)-V(I-1,J))/DX2                                    00054800
54900           DVDY=(Y(I,J+1)-Y(I,J))/DY                                       00054900
55000           DTDX=(Z(I+1,J)-Z(I-1,J))/DX2                                    00055000
55100           DTDY=(Z(I,J+1)-Z(I,J-1))/DY2                                    00055100
55200      501  SS2=SI(I,J)**2                                                  00055200
55300           CC2=CO(I,J)**2                                                  00055300
55400            CH=COSH(RKA(I,J)*D(I,J-1))                                     00055400
55500           IF(ID.EQ.1) GO TO 301                                           00055500
55600           X=0.0                                                           00055600
55700           GO TO 302                                                       00055700
55800      301  X=RKA(I,J)*CF*SIGMA*H(I,J)/(6.*PI*(CH**3.-CH))                 00055800
55900           X=X+G*PERM*RKA(1,J)/(VISCOS*CH*CH)                              00055900
56100      302  NA=N1                                                           00056100
56200           NB=N2                                                           00056200
56300           SIGXX=(2.0*FF-0.5)*CC2+(FF-0.5)*SS2                             00056300
56400           SIGYY=(2.0*FF-0.5)*SS2+(FF-0.5)*CC2                             00056400
```

```
56500           SIGX(I,J) = SIGXX                                                00056500
56600           SIGY(I,J) = SIGYY                                                00056600
56700           TAUXY=FF*SI(I,J)*CO(I,J)                                         00056700
56800           SIGXY(I,J)=TAUXY                                                 00056800
56900           S(I,J)=CG(I,J)*(SI(I,J)*DTDX-CO(I,J)*DTDY)-(DUDX+DVDY)-(CO(I,J)*DC 00056900
57000          *GDX+SI(I,J)*DCGDY)-(SIGXX*DUDX+TAUXY*DUDY+TAUXY*DVDX+SIGYY*DVDY)-X 00057000
57100           GO TO 500                                                        00057100
57200       499 H(I,J)=0.0                                                       00057200
57300           SIGXY(I,J)=0.0                                                   00057300
57400       500 CONTINUE                                                         00057400
57500           DO 510 II=1,M2                                                   00057500
57600           I=M-II                                                           00057600
57700           DO 580 ITT=1,ITMAX                                               00057700
57800           IFLAG=1                                                          00057800
57900           DO 520 J=2,N1                                                    00057900
58000           IF(D(I,J-1) .LE. DD) GO TO 520                                   00058000
58100           IB(I,J)=1                                                        00058100
58200           CC1=(V(I,J)+CG(I,J)*SI(I,J))/DY                                  00058200
58300           CC2=(U(I,J)+CG(I,J)*CO(I,J))/DX                                  00058300
58400           HNEW(I,J)= (CC1*H(I,J-1)-CC2*H(I+1,J))/(CC1-CC2-S(I,J)/2.0)      00058400
58500           IF (HNEW(I,J) .LT. 0.0) GO TO 810                                00058500
58600           IF(HNEW(I,J).LE.HBREAK(I,J)) GO TO 850                           00058600
58700       810 HNEW(I,J)=HBREAK(I,J)                                            00058700
58800           IB(I,J)=0                                                        00058800
58900       850 CONTINUE                                                         00058900
59000           IF(ABS(HNEW(I,J)-H(I,J)).GT.(EPSH*ABS(HNEW(I,J)))) IFLAG=0       00059000
59100       C*      BOUNDARY CONDITIONS                                          00059100
59200           IF(J.NE.2) GO TO 800                                             00059200
59300           HNEW(I,N1)=HNEW(I,J)                                             00059300
59400           IB(I,N1)=IB(I,J)                                                 00059400
59500           GO TO 520                                                        00059500
59600       800 IF(J.NE.3) GO TO 801                                             00059600
59700           HNEW(I,N2)=HNEW(I,J)                                             00059700
59800           IB(I,N2)=IB(I,J)                                                 00059800
59900       801 IF(J.NE.N) GO TO 802                                             00059900
60000           HNEW(I,1)=HNEW(I,J)                                              00060000
60100           IB(I,1)=IB(I,J)                                                  00060100
60200           GO TO 520                                                        00060200
60300       802 IF(J.NE.N1) GO TO 520                                            00060300
60400           HNEW(I,2)=HNEW(I,J)                                              00060400
60500           IB(I,2)=IB(I,J)                                                  00060500
60600       520 CONTINUE                                                         00060600
60700           DO 625 J=1,N2                                                    00060700
60800           H(I,J)=HNEW(I,J)                                                 00060800
60900       625 CONTINUE                                                         00060900
61000           IF(IFLAG.EQ.1) GO TO 570                                         00061000
61100       580 CONTINUE                                                         00061100
61200           WRITE(6,540) I,ITT                                               00061200
61300       540 FORMAT(' RELAXATION FOR THE WAVE HEIGHT FAILED TO CONVERGE',     00061300
61400          *'ON ROW ',I5,'AFTER',I6,'ITERATIONS')                            00061400
61500           WRITE(6,541)(H(I,J),J=1,N2    )                                  00061500
61600       541 FORMAT('  LAST VALUES OF H ARE'/,10F13.5)                        00061600
61700       C       RETURN                                                       00061700
61800       570 CONTINUE                                                         00061800
61900       510 CONTINUE                                                         00061900
62000           RETURN                                                           00062000
62100           END                                                              00062100
62200       C*---------------------------------------------------------------    00062200
62300       C*                                                                   00062300
62400           SUBROUTINE ANGLE(ITMAX)                                          00062400
62500       C*                                                                   00062500
62600       C*      ANGLE CALCULATES THE LOCAL WAVE DIRECTION AT EACH GRID LOCATION 00062600
```

```
62700     C*                                                                      00062700
62800     C*----------------------------------------------------------------------00062800
62900           COMMON D(50,50),W(50,50),Y(50,50),Z(50,50),SI(50,50),CO(50,50),   00062900
63000          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)   00063000
63100          *,DDDY(50,50),U(50,50),V(50,50)                                    00063100
63200           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,    00063200
63300          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                         00063300
63400     C*----------------------------------------------------------------------00063400
63500     C*    DEFINE STATEMENT FUNCTIONS C,SS,DUDX,DUDY,DVDX,DVDY,DKDX          00063500
63600     C*    DKDY,FAC,F                                                        00063600
63700     C*----------------------------------------------------------------------00063700
63800           C(I,J)=0.25*(CO(I+1,J)+CO(I-1,J)+CO(I,J+1)+CO(I,J-1))+0.125*((Z(I 00063800
63900          *+1,J)-Z(I-1,J))*(SI(I+1,J)-SI(I-1,J))+ (Z(I,J+1)-Z(I,J-1))*(SI(I, 00063900
64000          *J+1)-SI(I,J-1)))                                                  00064000
64100           SS(I,J)=0.25*(SI(I+1,J)+SI(I-1,J)+SI(I,J+1)+SI(I,J-1))+0.125*((Z(I 00064100
64200          *+1,J)-Z(I-1,J))*(CO(I-1,J)-CO(I+1,J)) + (Z(I,J+1)-Z(I,J-1))*(CO(I, 00064200
64300          *J-1)-CO(I,J+1)))                                                  00064300
64400           DUDX(I,J)=(W(I+1,J)-W(I,J))/DX                                    00064400
64500           DUDY(I,J)=(U(I,J+1)-U(I,J-1))/DY2                                 00064500
64600           DVDX(I,J)=(V(I+1,J)-V(I-1,J))/DX2                                 00064600
64700           DVDY(I,J)=(Y(I,J+1)-Y(I,J))/DY                                    00064700
64800           F(I,J)=U(I,J)*COSI+V(I,J)*SINI + 0.5*A*(1.0 + ARG2/SINH2)/RK      00064800
64900           DKDY(I,J)=(-(COSI*DUDY(I,J) + SINI*DVDY(I,J))-A*DDDY(I,J-1)/SINH2) 00064900
65000          */FF                                                               00065000
65100           DKDX(I,J)=(-(COSI*DUDX(I,J)+SINI*DVDX(I,J))-A*DDDX(I,J-1)/SINH2)/F 00065100
65200          *F                                                                 00065200
65300           FAC(I,J)=U(I,J)*SINI-V(I,J)*COSI                                  00065300
65400           DO 200 ITT=1,ITMAX                                                00065400
65500           MWET=M-IWET                                                       00065500
65600           IFLAG=1                                                           00065600
65700           DO 210 II=1,MWET                                                  00065700
65800           I=M-II                                                            00065800
65900           DO 210 J=2,N1                                                     00065900
66000           IF(D(I,J-1).LE.DD) GO TO 210                                      00066000
66100           COSI=C(I,J)                                                       00066100
66200           SINI=SS(I,J)                                                      00066200
66300           JJ=J-1                                                            00066300
66400           CALL WVNUM(D(I,JJ),COSI,SINI,U(I,J),V(I,J),RK,A)                  00066400
66500           ARG2=2.0*RK*D(I,JJ)                                               00066500
66600           SINH2=SINH(ARG2)                                                  00066600
66700           FF=F(I,J)                                                         00066700
66800           IF(FF.GT.0.0) GO TO 450                                           00066800
66900           WRITE(6,451) I,J,D(I,JJ),COSI,SINI,U(I,J),V(I,J),RK,A             00066900
67000       451 FORMAT(10X'FF IS NEGATIVE--OUTPUT I,J,D,COSI,SINI,U,V,RK,A'/,     00067000
67100          *2I5,7E13.4)                                                       00067100
67200           GO TO 210                                                         00067200
67300       450 FACI=FAC(I,J)                                                     00067300
67400           DEN1=(SINI-COSI*FACI/FF)/DY                                       00067400
67500           DEN2=(COSI+SINI*FACI/FF)/DX                                       00067500
67600           DEN=DEN1-DEN2                                                     00067600
67700           ZNEW=(COSI*DKDY(I,J)-SINI*DKDX(I,J)+Z(I,J-1)*DEN1-Z(I+1,J)*       00067700
67800          *DEN2)/DEN                                                         00067800
67900           IF(ABS(ZNEW-Z(I,J)).GT.(EPSA*ABS(ZNEW))) IFLAG=0                  00067900
68000           Z(I,J)=ZNEW                                                       00068000
68100       992 CO(I,J)=COS(Z(I,J))                                               00068100
68200           SI(I,J)=SIN(Z(I,J))                                               00068200
68300           IF(J.NE.2) GO TO 400                                              00068300
68400           Z(I,N1)=Z(I,2)                                                    00068400
68500           CO(I,N1)=CO(I,2)                                                  00068500
68600           SI(I,N1)=SI(I,2)                                                  00068600
68700           GO TO 210                                                         00068700
68800       400 IF(J.NE.3) GO TO 401                                              00068800
```

```
68900           Z(I,N2)=Z(I,3)                                                  00068900
69000           CO(I,N2)=CO(I,3)                                                00069000
69100           SI(I,N2)=SI(I,3)                                                00069100
69200           GO TO 210                                                       00069200
69300       401 IF(J.NE.N) GO TO 402                                            00069300
69400           Z(I,1)=Z(I,N)                                                   00069400
69500           CO(I,1)=CO(I,N)                                                 00069500
69600           SI(I,1)=SI(I,N)                                                 00069600
69700           GO TO 210                                                       00069700
69800       402 CONTINUE                                                        00069800
69900           IF(J.NE.N1) GO TO 210                                           00069900
70000           Z(I,2)=Z(I,N1)                                                  00070000
70100           CO(I,2)=CO(I,N1)                                                00070100
70200           SI(I,2)=SI(I,N1)                                                00070200
70300       210 CONTINUE                                                        00070300
70400           IF(IFLAG.EQ.1) GO TO 250                                        00070400
70500       200 CONTINUE                                                        00070500
70600           WRITE(6,220) ITMAX                                              00070600
70700       220 FORMAT(" RELAXATION FOR THETA FAILED AFTER",I4,3X,              00070700
70800          *"ITERATIONS")                                                   00070800
70900           STOP                                                            00070900
71000       250 CONTINUE                                                        00071000
71100           RETURN                                                          00071100
71200           END                                                             00071200
71300       C*--------------------------------------------------------------------00071300
71400       C*                                                                  00071400
71500           SUBROUTINE WVNUM(D,COSI,SINI,U,V,RK,A)                          00071500
71600       C*                                                                  00071600
71700       C*    SUBROUTINE WVNUM COMPUTES WAVE NUMBER INCLUDING WAVE-         00071700
71800       C*    CURRENT INTERACTION.                                          00071800
71900       C*                                                                  00071900
72000       C*--------------------------------------------------------------------00072000
72100           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00072100
72200          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00072200
72300           EPSK=0.001                                                      00072300
72400           RK=PI2/(T*SQRT(G*D))                                            00072400
72500           DO 100I=1,50                                                    00072500
72600           A=SIGMA-U*RK*COSI-RK*V*SINI                                     00072600
72700           A2=A**2                                                         00072700
72800           ARG=RK*D                                                        00072800
72900           F1=EXP(ARG)                                                     00072900
73000           F2=1.0/F1                                                       00073000
73100           SECH=2.0/(F1+F2)                                                00073100
73200           SECH2=SECH**2                                                   00073200
73300           TT=TANH(ARG)                                                    00073300
73400           FK=G*RK*TT-A2                                                   00073400
73500           FFK=G*(ARG*SECH2+TT)+2.0*(U*COSI+V*SINI)*A                      00073500
73600           RKNEW=RK-FK/FFK                                                 00073600
73700           IF(ABS(RKNEW-RK).LE.(ABS(EPSK*RKNEW)))GO TO 110                 00073700
73800           RK=RKNEW                                                        00073800
73900       100 CONTINUE                                                        00073900
74000           WRITE(6,101)I,RK,T,D,U,V                                        00074000
74100       101 FORMAT(" INTERATION FOR K FAILED TO CONVERGE: I,K,D,U,V"        00074100
74200          *,I6,5F10.5)                                                     00074200
74300           CALL EXIT                                                       00074300
74400           RETURN                                                          00074400
74500       110 RK=RKNEW                                                        00074500
74600           A=SIGMA-RK*U*COSI-RK*V*SINI                                     00074600
74700       C*--------------------------------------------------------------------00074700
74800       C*    CHECK IF RK NEGATIVE, THIS CONDITION CAN ARISE IF LOCAL CURRENT 00074800
74900       C*    IS TOO STRONG                                                 00074900
75000       C*--------------------------------------------------------------------00075000
```

```
75100          IF(RK.GT.0.0)GO TO 120                                          00075100
75200          WRITE(6,130)D,COSI,SINI,U,V,RK,A                                00075200
75300      130 FORMAT(" RK IS NEGATIVE: D,COSI,SINI,U,V,RK,A",                 00075300
75400         *7F10.5)                                                         00075400
75500          CALL EXIT                                                       00075500
75600      120 RETURN                                                          00075600
75700          END                                                             00075700
75800      C*--------------------------------------------------------------   00075800
75900      C*                                                                  00075900
76000          SUBROUTINE SNELL (THETAO,HH,ITER)                               00076000
76100      C*                                                                  00076100
76200      C*    SNELL CALCULATES THE FIRST GUESS OF THE REFRACTION ANGLE      00076200
76300      C*    AT EACH GRID, AND ALSO THE WAVE HEIGHT AT THE LAST GRID(M)    00076300
76400      C*    FOR EACH ITERATION THAT THE DEEPWATER WAVE IS BUILDING UP.    00076400
76500      C*                                                                  00076500
76600      C*------------------------------------------------------------------00076600
76700          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50), 00076700
76800         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50) 00076800
76900         *,DDDY(50,50)                                                    00076900
77000          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00077000
77100         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00077100
77200          I=IDRY                                                          00077200
77300       30 I=I+1                                                           00077300
77400          IF(ITER.GT.1) I=M                                               00077400
77500          DO 600 J=2,N1                                                   00077500
77600          CALL WVNUM(D(I,J-1),-1.0,0.0,0.0,0.0,0.0,RK,A)                  00077600
77700          AA = RK* D(I,J-1)                                               00077700
77800          ANG = ARSIN(SIN(THETAO)*TANH(AA))                               00077800
77900          ANG = PI-ANG                                                    00077900
78000          ARG = 2.0*AA                                                    00078000
78100          SHOAL = SQRT(1.0/(TANH(AA)*(1.0+ARG/SINH(ARG))))                00078100
78200          REF = SQRT(COS(THETAO)/COS(ANG))                                00078200
78300          WVHT = HH*SHOAL*REF                                             00078300
78400          HB = 0.12*TANH(AA)*PI2/RK                                       00078400
78500          IF(WVHT .GT. HB) WVHT = HB                                      00078500
78600          IN = 1                                                          00078600
78700          IF(WVHT .GT. HB) IN = 0                                         00078700
78800          SS = SIN(ANG)                                                   00078800
78900          CC = COS(ANG)                                                   00078900
79000       43 CONTINUE                                                        00079000
79100          H(I,J) = WVHT                                                   00079100
79200          IB(I,J) = IN                                                    00079200
79300          Z(I,J) = ANG                                                    00079300
79400          SI(I,J) = SS                                                    00079400
79500          CO(I,J) = CC                                                    00079500
79600      600 CONTINUE                                                        00079600
79700          IF(I.EQ.M) GO TO 85                                             00079700
79800          GO TO 30                                                        00079800
79900       85 CONTINUE                                                        00079900
80000      C*--------------------------------------------------------------   00080000
80100      C*    BOUNDARY CONDITIONS                                           00080100
80200      C*--------------------------------------------------------------   00080200
80300          L = 1                                                           00080300
80400          K = N                                                           00080400
80500          J = 1                                                           00080500
80600       45 DO 60 I = 1,M                                                   00080600
80700          H(I,J) = H(I,K)                                                 00080700
80800          IB (I,J) = IB(I,K)                                              00080800
80900          Z(I,J)=Z(I,K)                                                   00080900
81000          SI(I,J) = SI(I,K)                                               00081000
81100       60 CO(I,J) = CO(I,K)                                               00081100
81200          L = L+1                                                         00081200
```

```
81300           IF (L .GT. 3) RETURN                                                00081300
81400           IF (L .EQ. 3)  GO TO 610                                            00081400
81500           L = 2                                                               00081500
81600           J = 2                                                               00081600
81700           K=N1                                                                00081700
81800           GO TO 45                                                            00081800
81900       610 J = N2                                                              00081900
82000           K = 3                                                               00082000
82100           GO TO 45                                                            00082100
82200           RETURN                                                              00082200
82300           END                                                                 00082300
82400     C*--------------------------------------------------------------------     00082400
82500     C*                                                                        00082500
82600           SUBROUTINE TAUSB (WIND, WINANG, CF,ITO)                             00082600
82700     C*                                                                        00082700
82800     C*    THIS SUBROUTINE CALCULATES SURFACE SHEAR STRESS BY VAN DORN'S       00082800
82900     C*    METHOD, AND AVERAGE BOTTOM SHEAR STRESS BY THE METHOD OF            00082900
83000     C*    DALRYMPLE AND LIU                                                   00083000
83100     C*                                                                        00083100
83200     C*    WIND SPEED MUST BE IN METERS/SECOND                                 00083200
83300     C*                                                                        00083300
83400     C*--------------------------------------------------------------------     00083400
83500           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),     00083500
83600          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),    00083600
83700          *DDDY(50,50),W(50,50),Y(50,50)                                       00083700
83800           COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),             00083800
83900          *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)          00083900
84000           COMMON/STRESS/SIGXX(50,50),SIGYY(50,50),SIGXY(50,50),TAUBX(50,50),  00084000
84100          *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                              00084100
84200           COMMON/REF/ZZ(50,50),HNEW(50,50),RKA(50,50)                         00084200
84300           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,      00084300
84400          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                           00084400
84500           DIMENSION B(40),F(40),GG(40)                                        00084500
84600     C*--------------------------------------------------------------------     00084600
84700     C*    COMPUTE SURFACE SHEAR STRESS                                        00084700
84800     C*--------------------------------------------------------------------     00084800
84900           IF(IT.GT.ITO+1) GO TO 1                                             00084900
85000           VANCON=1.1E-06                                                      00085000
85100           IF (WIND .GE.7.2) VANCON = 1.1E-06+2.5E-06*((1.-7.2/WIND)**2.)      00085100
85200           WINDX = WIND*COS(WINANG)                                            00085200
85300           WINDY = WIND*SIN(WINANG)                                            00085300
85400           CON1  = RHO*VANCON*ABS(WIND)                                        00085400
85500           DO 20 J = 1,N2                                                      00085500
85600           DO 20 I = 2, M                                                      00085600
85700           TAUSX(I,J) =- CON1 *WINDX                                           00085700
85800           TAUSY(I,J) = CON1 *WINDY                                            00085800
85900        20 CONTINUE                                                          00085900
86000     C*--------------------------------------------------------------------     00086000
86100     C*    COMPUTE BOTTOM SHEAR STRESS                                         00086100
86200     C*--------------------------------------------------------------------     00086200
86300        1  L=16                                                               00086300
86400           DELTA=PI2/L                                                         00086400
86500           VALUE=(RHO*CF)/(24.0*L)                                             00086500
86600           DO 4 J=2,N1                                                         00086600
86700           K=J-1                                                               00086700
86800           DO 4 I=IWET,M1                                                      00086800
86900           PHI=Z(I,J)-PI                                                       00086900
87000           CC=COS(PHI)                                                         00087000
87100           SS=SIN(PHI)                                                         00087100
87200           UMAX=(PI*H(I,J))/(T*SINH(RKA(I,J)*D(I,K)))                          00087200
87300           EXPR1=UMAX*SS                                                       00087300
87400           EXPR2=UMAX*CC                                                       00087400
```

```
87500          EXPR3=2.0*EXPR2                                                  00087500
87600          EXPR4=2.0*EXPR1                                                  00087600
87700          WM1=(UM1(I,J)+UM1(I+1,J))/2.0                                    00087700
87800          YM1=(VM1(I,J)+VM1(I,J+1))/2.0                                    00087800
87900          EXPR5=WM1**2                                                     00087900
88000          EXPR6=YM1**2                                                     00088000
88100          DELX=0.0                                                         00088100
88200          DO 2 LL=1,L+1,1                                                  00088200
88300          B(LL)=SQRT(EXPR5+EXPR6+(UMAX*COS(DELX))**2+COS(DELX)*            00088300
88400         *(WM1*EXPR3+YM1*EXPR4))                                           00088400
88500          GG(LL)=(YM1+EXPR1*COS(DELX))*B(LL)                               00088500
88600          F(LL)=(WM1+EXPR2*COS(DELX))*B(LL)                                00088600
88700          DELX=DELX+DELTA                                                  00088700
88800        2 CONTINUE                                                         00088800
88900          SUM=0.0                                                          00088900
89000          SUMM=0.0                                                         00089000
89100          DO 3 MM=2,L,2                                                    00089100
89200          SUM=SUM+F(MM-1)+4.0*F(MM)+F(MM+1)                                00089200
89300          SUMM=SUMM+GG(MM-1)+4.0*GG(MM)+GG(MM+1)                           00089300
89400        3 CONTINUE                                                         00089400
89500          TAUBX(I,J)=SUM*VALUE                                             00089500
89600          TAUBY(I,J)=SUMM*VALUE                                            00089600
89700        4 CONTINUE                                                         00089700
89800        42 DO 5 I=1,M                                                      00089800
89900          TAUBX(I,1) = TAUBX(I,N)                                          00089900
90000          TAUBX(I,N1) = TAUBX(I,2)                                         00090000
90100          TAUBX(I,N2) = TAUBX(I,3)                                         00090100
90200          TAUBY(I,1) = TAUBY(I,N)                                          00090200
90300          TAUBY(I,N1) = TAUBY(I,2)                                         00090300
90400          TAUBY(I,N2) = TAUBY(I,3)                                         00090400
90500        5 CONTINUE                                                         00090500
90600          RETURN                                                           00090600
90700          END                                                              00090700
90800    C*--------------------------------------------------------------------00090800
90900    C*                                                                     00090900
91000          SUBROUTINE DGRAD                                                 00091000
91100    C*                                                                     00091100
91200    C*    CALCULATE THE SPATIAL GRADIENTS IN TOTAL DEPTH AFTER EACH        00091200
91300    C*    UPDATING OF THE TOTAL DEPTH                                      00091300
91400    C*                                                                     00091400
91500    C*--------------------------------------------------------------------00091500
91600          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CD(50,50),  00091600
91700         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50)  00091700
91800         *,DDDY(50,50)                                                     00091800
91900          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,   00091900
92000         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                        00092000
92100          DO 60 I = 2,M1                                                   00092100
92200          DO 60 J = 3,N1                                                   00092200
92300          DDDX(I,J) =-(D(I-1,J)-D(I+1,J))/(2.*DX)                          00092300
92400          IF(J.EQ.N1) GO TO 61                                             00092400
92500          DDY=D(I,J-2)-8.0*D(I,J-1)+8.0*D(I,J+1)-D(I,J+2)                  00092500
92600          DDDY(I,J)=DDY/(12.0*DY)                                          00092600
92700          GO TO 60                                                         00092700
92800        61 DDY=D(I,N-1)-8.0*D(I,N)+8.0*D(I,N+2)-D(I,4)                     00092800
92900          DDDY(I,J)=DDY/(12.0*DY)                                          00092900
93000        60 CONTINUE                                                        00093000
93100          DO 70 I = 1,M                                                    00093100
93200          DDDX(I,1)=DDDX(I,N)                                              00093200
93300          DDDY(I,1)=DDDY(I,N)                                              00093300
93400          DDDX(I,2)=DDDX(I,N1)                                             00093400
93500          DDDY(I,2)=DDDY(I,N1)                                             00093500
93600          DDDX(I,N2)=DDDX(I,3)                                             00093600
```

```
93700          DDDY(I,N2)=DDDY(I,3)                                                    00093700
93800       70 CONTINUE                                                               00093800
93900          RETURN                                                                 00093900
94000          END                                                                    00094000
94100    C*-------------------------------------  ------------------------------     00094100
94200    C*                                                                          00094200
94300          SUBROUTINE MOMEN                                                       00094300
94400    C*                                                                          00094400
94500    C*     MOMEN CALCULATES VELOCITIES USING THE FULL NONLINEAR MOMENTUM        00094500
94600    C*     EQUATIONS                                                            00094600
94700    C*                                                                          00094700
94800    C*-------------------------------------------------------------------------00094800
94900          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),      00094900
95000         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),     00095000
95100         *DDDY(50,50),W(50,50),Y(50,50)                                        00095100
95200          COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),              00095200
95300         *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)           00095300
95400          COMMON/STRESS/SIGX(50,50),SIGY(50,50),SIGXY(50,50),TAUBX(50,50),     00095400
95500         *TAUBY(50,50),TAUSX(50,50),TAUSY(50,50)                               00095500
95600          COMMON/EDDY/EX(50,50),EY,VCON                                        00095600
95700          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,       00095700
95800         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                            00095800
95900          DO 97 I=IWET,M                                                       00095900
96000          DO 96 J=3,N1                                                         00096000
96100          EX(I,J)=((I-IWET+1)*DX)*VCON*SQRT(G*DM1(I,J-1))                      00096100
96200          IF(I.GE.12) EX(I,J)=.875                                             00096200
96300       96 CONTINUE                                                             00096300
96400          EX(I,N2)=EX(I,3)                                                     00096400
96500          EX(I,2)=EX(I,N1)                                                     00096500
96600          EX(I,1)=EX(I,N)                                                      00096600
96700       97 CONTINUE                                                             00096700
96800       99 DO 21 J=3,N1                                                         00096800
96900          K = J-1                                                              00096900
97000          DO 21 I=IWET,M1                                                      00097000
97200          UP1(I,J)=(((D(I+1,K)+D(I,K))*U(I+1,J)+(D(I,K)+D(I-1,K))*             00097200
97300         *U(I,J))*(U(I+1,J)+U(I,J))-((D(I,K)+D(I-1,K))*U(I,J)+                 00097300
97400         *(D(I-1,K)+D(I-2,K))*U(I-1,J))*(U(I,J)+U(I-1,J)))/(8.0*DX)            00097400
97500          UP1(I,J)=UP1(I,J)+(((D(I,K+1)+D(I,K))*V(I,J+1)+(D(I-1,K+1)+          00097500
97600         *D(I-1,K))*V(I-1,J+1))*(U(I,J)+U(I,J+1))-((D(I,K)+D(I,K-1))*          00097600
97700         * V(I,J)+(D(I-1,K)+D(I-1,K-1))*V(I-1,J))*(U(I,J)+U(I,J-1)))           00097700
97800         */(8.0*DY)                                                           00097800
97900          UP1(I,J)=UP1(I,J)+(D(I,K)+D(I-1,K))*(ETA(I,J)-ETA(I-1,J))            00097900
98000         **(G/(DX*2.0))                                                       00098000
98100          UP1(I,J)=UP1(I,J)-(TAUSX(I,J)+TAUSX(I-1,J))/(2.0*RHO)                00098100
98200          UP1(I,J)=UP1(I,J)+(TAUBX(I,J)+TAUBX(I-1,J))/(2.0*RHO)                00098200
98300          UP1(I,J)=UP1(I,J)+(SIGX(I,J)-SIGX(I-1,J))/(DX*RHO)+(SIGXY(I,         00098300
98400         *J+1)-SIGXY(I,J-1)+SIGXY(I-1,J+1)-SIGXY(I-1,J-1))/(4.0*DY*RHO)        00098400
98500          UP1(I,J)=UP1(I,J)-(DM1(I,K)+DM1(I-1,K))*((UM1(I,J+1)-                00098500
98600         *UM1(I,J))*EY-(UM1(I,J)-UM1(I,J-1))*EY)/(2.0*DY*DY)                   00098600
98700          UP1(I,J)=UP1(I,J)-(DM1(I,K)+DM1(I-1,K))*((VM1(I,J+1)-                00098700
98800         *VM1(I-1,J+1))*(EX(I,J+1)+EX(I,J)+EX(I-1,J)+EX(I-1,J+1))-             00098800
98900         *(VM1(I,J)-VM1(I-1,J))*(EX(I,J)+EX(I,J-1)+EX(I-1,J-1)+                00098900
99000         *EX(I-1,J)))/(8.0*DX*DY)                                             00099000
99100          VP1(I,J)=(((D(I+1,K)+D(I,K))*U(I+1,J)+(D(I+1,K-1)+D(I,K-1))*         00099100
99200         *U(I+1,J-1))*(V(I+1,J)+V(I,J))-((D(I,K)+D(I-1,K))*U(I,J)+             00099200
99300         *(D(I,K-1)+D(I-1,K-1))*U(I,J-1))*(V(I,J)+V(I-1,J)))/(8.0*DX)          00099300
99400          IF (K.EQ.2) GO TO 83                                                 00099400
99500          GO TO 84                                                             00099500
99600       83 VP1(I,J)=VP1(I,J)+(((D(I,K+1)+D(I,K))*V(I,J+1)+(D(I,K)+            00099600
99700         *D(I,K-1))*V(I,J))*(V(I,J+1)+V(I,J))-((D(I,K)+D(I,K-1))*             00099700
99800         *V(I,J)+(D(I,K-1)+D(I,N-1))*V(I,J-1))*(V(I,J)+V(I,J-1)))/(8.0*DY)      00099800
99900          GO TO 85                                                             00099900
```

```
100000       84 VP1(I,J)=VP1(I,J)+(((D(I,K+1)+D(I,K))*V(I,J+1)+(D(I,K)+                00100000
100100         *D(I,K-1))*V(I,J))*(V(I,J+1)+V(I,J))-((D(I,K)+D(I,K-1))*               00100100
100200         *V(I,J)+(D(I,K-1)+D(I,K-2))*V(I,J-1))*(V(I,J)+V(I,J-1)))/(8.0*DY)      00100200
100300       85 VP1(I,J)=VP1(I,J)+(D(I,K)+D(I,K-1))*(ETA(I,J)-ETA(I,J-1))*           00100300
100400         *(G/(2.0*DY))                                                         00100400
100500          VP1(I,J)=VP1(I,J)-(TAUSY(I,J)+TAUSY(I,J-1))/(2.0*RHO)                00100500
100600          VP1(I,J)=VP1(I,J)+(TAUBY(I,J)+TAUBY(I,J-1))/(2.0*RHO)                00100600
100700          VP1(I,J)=VP1(I,J)+(SIGXY(I+1,J-1)-SIGXY(I-1,J-1)+SIGXY               00100700
100800         *(I+1,J)-SIGXY(I-1,J))/(4.0*RHO*DX)+(SIGY(I,J)-SIGY(I,J-1))           00100800
100900         */(RHO*DY)                                                            00100900
101000          VP1(I,J)=VP1(I,J)-(DM1(I,K)+DM1(I,K-1))*((EX(I+1,J-1)+               00101000
101100         *EX(I+1,J)+EX(I,J-1)+EX(I,J))*(VM1(I+1,J)-VM1(I,J))-(                 00101100
101200         *EX(I,J-1)+EX(I,J)+EX(I-1,J-1)+EX(I-1,J))*(VM1(I,J)-                  00101200
101300         *VM1(I-1,J)))/(8.0*DX*DX)                                             00101300
101400          VP1(I,J)=VP1(I,J)-(DM1(I,K)+DM1(I,K-1))*((UM1(I+1,J)-                00101400
101500         *UM1(I+1,J-1))*EY-(UM1(I,J)-UM1(I,J-1))*EY)/(2.0*DY*DX)               00101500
101600       21 CONTINUE                                                           00101600
101700          RETURN                                                             00101700
101800          END                                                                00101800
101900    C*-------------------------------------------------------------------    00101900
102000    C*                                                                       00102000
102100          SUBROUTINE ETAS(RETA,DELT)                                         00102100
102200    C*                                                                       00102200
102300    C*     ETAS CALCULATES THE CHANGE IN MEAN WATER LEVEL DUE TO WAVE ACTION  00102300
102400    C*                                                                       00102400
102500    C*-------------------------------------------------------------------    00102500
102600          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),    00102600
102700         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50),   00102700
102800         *DDDY(50,50),W(50,50),Y(50,50)                                      00102800
102900          COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),            00102900
103000         *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)         00103000
103100          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,     00103100
103200         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                          00103200
103300          DIMENSION RETA(50,50)                                              00103300
103400          DO 20 J=2,N1                                                       00103400
103500          DO 20 I=IWET,M1                                                    00103500
103600          K=J-1                                                              00103600
103700          ETAP1(I,J)=RETA(I,J)-(DELT*0.5)*ETAP1(I,J)                         00103700
103800       22 DP1(I,K)=D(I,K)-ETA(I,J)+ETAP1(I,J)                                00103800
103900       20 CONTINUE                                                           00103900
104000          DO 23 J=2,N1                                                       00104000
104100          K=J-1                                                              00104100
104200          DO 24 I=1,IDRY                                                     00104200
104300          DP1(I,K)=D(I,K)                                                    00104300
104400       24 CONTINUE                                                           00104400
104500          DP1(M,K)=D(M,K)                                                    00104500
104600       23 CONTINUE                                                           00104600
104700          DO 60 I=1,M                                                        00104700
104800          ETAP1(I,N2)=ETAP1(I,3)                                             00104800
104900          ETAP1(I,N1)=ETAP1(I,2)                                             00104900
105000          ETAP1(I,1)=ETAP1(I,N)                                              00105000
105100          DP1(I,N2)=DP1(I,3)                                                 00105100
105200          DP1(I,N1)=DP1(I,2)                                                 00105200
105300          DP1(I,1)=DP1(I,N)                                                  00105300
105400       60 CONTINUE                                                           00105400
105500          DO 92 I=1,M                                                        00105500
105600          IF(DP1(I,1).GT.DD) GO TO 93                                        00105600
105700       92 CONTINUE                                                           00105700
105800       93 IWET=I                                                             00105800
105900          IDRY=IWET-1                                                        00105900
106000          RETURN                                                             00106000
106100          END                                                                00106100
```

```
106200    C*--------------------------------------------------------------------  00106200
106300    C*                                                                     00106300
106400          SUBROUTINE UCALC(TD,TU,TV,DELT)                                  00106400
106500    C*                                                                     00106500
106600    C*    UCALC CALCULATES THE DEPTH AVERAGED VELOCITIES BASED ON THE      00106600
106700    C*    RESULTS OF SUBROUTINE MOMEN                                      00106700
106800    C*                                                                     00106800
106900    C*--------------------------------------------------------------------  00106900
107000          COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50),  00107000
107100         *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50), 00107100
107200         *DDDY(50,50),W(50,50),Y(50,50)                                    00107200
107300          COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),          00107300
107400         *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)       00107400
107500          COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,   00107500
107600         *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                        00107600
107700          DIMENSION  TD(50,50),TU(50,50),TV(50,50)                         00107700
107800          DO 24 I=IWET,M1                                                  00107800
107900          DO 24 J=3,N1                                                     00107900
108000          K=J-1                                                            00108000
108100          UP1(I,J)=((TD(I,K)+TD(I-1,K))*TU(I,J))/(DP1(I,K)+DP1(I-1,        00108100
108200         *K))-((DELT*2.0)/(DP1(I,K)+DP1(I-1,K)))*UP1(I,J)                  00108200
108300          VP1(I,J)=((TD(I,K)+TD(I,K-1))*TV(I,J))/(DP1(I,K)+                00108300
108400         *DP1(I,K-1))-((DELT*2.0)/(DP1(I,K)+DP1(I,K-1)))*VP1(I,J)          00108400
108500       24 CONTINUE                                                         00108500
108600          DO 22 J=3,N1                                                     00108600
108700          DO 27 I=1,IWET                                                   00108700
108800          UP1(I,J)=0.0                                                     00108800
108900       27 CONTINUE                                                         00108900
109000          UP1(M,J)=0.0                                                     00109000
109100          VP1(M,J)=VP1(M1,J)                                               00109100
109200       22 CONTINUE                                                         00109200
109300          DO 60 I=1,M                                                      00109300
109400          UP1(I,N2)=UP1(I,3)                                               00109400
109500          UP1(I,2)=UP1(I,N1)                                               00109500
109600          UP1(I,1)=UP1(I,N)                                                00109600
109700          VP1(I,N2)=VP1(I,3)                                               00109700
109800          VP1(I,2)=VP1(I,N1)                                               00109800
109900          VP1(I,1)=VP1(I,N)                                                00109900
110000       60 CONTINUE                                                         00110000
110100          DO 69 J=2,N1                                                     00110100
110200          DO 68 I=1,M1                                                     00110200
110300          W(I,J)=(UP1(I,J)+UP1(I+1,J))/2.0                                 00110300
110400          Y(I,J)=(VP1(I,J)+VP1(I,J+1))/2.0                                 00110400
110500       68 CONTINUE                                                         00110500
110600          Y(M,J)=0.0                                                       00110600
110700       69 CONTINUE                                                         00110700
110800          DO 79 I=1,M                                                      00110800
110900          W(I,N2)=W(I,3)                                                   00110900
111000          W(I,2)=W(I,N1)                                                   00111000
111100          W(I,1)=W(I,N)                                                    00111100
111200          Y(I,N2)=Y(I,3)                                                   00111200
111300          Y(I,2)=Y(I,N1)                                                   00111300
111400          Y(I,1)=Y(I,N)                                                    00111400
111500       79 CONTINUE                                                         00111500
111600          RETURN                                                           00111600
111700          END                                                              00111700
111800    C*--------------------------------------------------------------------  00111800
111900    C*                                                                     00111900
112000          SUBROUTINE ROLBAC                                                00112000
112100    C*                                                                     00112100
112200    C*    ROLBAC UPDATES THE VALUES OF ETA,U,V,AND D FOR THE NEXT          00112200
112300    C*    ITERATION                                                        00112300
```

```
112400     C*                                                                    00112400
112500     C*--------------------------------------------------------------------00112500
112600           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50), 00112600
112700          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50), 00112700
112800          *DDDY(50,50),W(50,50),Y(50,50)                                   00112800
112900           COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),         00112900
113000          *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)      00113000
113100           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00113100
113200          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00113200
113300           DO 1 I=1,M                                                      00113300
113400           DO 1 J=1,N2                                                     00113400
113500           ETAM1(I,J)=ETA(I,J)                                             00113500
113600           ETA(I,J)=ETAP1(I,J)                                             00113600
113700           UM1(I,J)=U(I,J)                                                 00113700
113800           U(I,J)=UP1(I,J)                                                 00113800
113900           VM1(I,J)=V(I,J)                                                 00113900
114000           V(I,J)=VP1(I,J)                                                 00114000
114100           D(I,J)=DP1(I,J)                                                 00114100
114200           DM1(I,J)=D(I,J)                                                 00114200
114300         1 CONTINUE                                                        00114300
114400           RETURN                                                          00114400
114500           END                                                             00114500
114600     C*--------------------------------------------------------------------00114600
114700     C*                                                                    00114700
114800           SUBROUTINE CONTIN                                               00114800
114900     C*                                                                    00114900
115000     C*    CONTIN CALCULATES THE FINAL CORRECTION TO TOTAL DEPTH BASED     00115000
115100     C*    ON THE EQUATION OF CONTINUITY                                   00115100
115200     C*                                                                    00115200
115300     C*--------------------------------------------------------------------00115300
115400           COMMON D(50,50),U(50,50),V(50,50),Z(50,50),SI(50,50),CO(50,50), 00115400
115500          *H(50,50),CG(50,50),S(50,50),HBREAK(50,50),IB(50,50),DDDX(50,50), 00115500
115600          *DDDY(50,50),W(50,50),Y(50,50)                                   00115600
115700           COMMON/VAL/DP1(50,50),DM1(50,50),UP1(50,50),UM1(50,50),         00115700
115800          *VP1(50,50),VM1(50,50),ETAP1(50,50),ETAM1(50,50),ETA(50,50)      00115800
115900           COMMON/CONST/ G,PI,PI2,RAD,EPSH,EPSA,DX,DY,DT,DX2,DY2,T,SIGMA,  00115900
116000          *M,N,N1,N2,M1,M2,AM,DD,IT,RHO,IWET,IDRY,ID                       00116000
116100           DO 20 J=2,N1                                                    00116100
116200           K=J-1                                                           00116200
116300           DO 20 I=IWET,M1                                                 00116300
116400           IF (K.EQ.1) GO TO 21                                            00116400
116500           ETAP1(I,J)=(((D(I+1,K)+D(I,K))*U(I+1,J)-                        00116500
116600          *(D(I,K)+D(I-1,K))*U(I,J))/DX+((D(I,K+1)+D(I,K))*V(I,J+1)-       00116600
116700          *(D(I,K)+D(I,K-1))*V(I,J))/DY)                                   00116700
116800           GO TO 20                                                        00116800
116900        21 ETAP1(I,J)=(((D(I+1,K)+D(I,K))*U(I+1,J)-                        00116900
117000          *(D(I,K)+D(I-1,K))*U(I,J))/DX+((D(I,K+1)+D(I,K))*V(I,J+1)        00117000
117100          *-(D(I,K)+D(I,N-1))*V(I,J))/DY)                                  00117100
117200        20 CONTINUE                                                        00117200
117300           RETURN                                                          00117300
117400           END                                                             00117400
```

Distribution

Office of Naval Research
Coastal Science Program
Code 42205
Arlington, VA 22217

Defense Documentation Center
Cameron Station
Alexandria, VA 22314

Director, Naval Research Lab.
ATTN: Technical Information Officer
Washington, D. C. 20375

Director
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, CA 91101

Chief of Naval Research
Code 100M
Office of Naval Research
Arlington, VA 22217

Office of Naval Research
Operational Applications Division
Code 200
Arlington, VA 22217

Office of Naval Research
Scientific Liaison Officer
Scripps Institution of Oceanography
La Jolla, CA 92093

Director Naval Research Laboratory
ATTN: Library, Code 2628
Washington, D. C. 20375

ONR Scientific Liaison Group
American Embassy - Room A-407
APO San Francisco, CA 96503

Commander
Naval Oceanographic Office
ATTN: Library, Code 1600
Washington, D. C. 20374

Naval Oceanographic Office
Code 3001
Washington, D. C. 20374

Chief of Naval Operations
OP 987P1
Department of the Navy
Washington, D. C. 20350

Oceanographer of the Navy
Hoffman II Building
200 Stovall Street
Alexandria, VA 22322

Naval Academy Library
U. S. Naval Academy
Annapolis, MD 21402

Commanding Officer
Naval Coastal Systems Laboratory
Panama City, FL 32401

Director
Coastal Engineering Research Center
U. S. Army Corps of Engineers
Kingman Building
Fort Belvoir, VA 22060

Officer in Charge
Environmental Research Prodctn Felty.
Naval Postgraduate School
Monterey, CA 93940

Director
Amphibious Warfare Board
U. S. Atlantic Fleet
Naval Amphibious Base
Norfolk, Little Creek, VA 23520

Commander, Amphibious Force
U. S. Pacific Fleet
Force Meteorologist
Comphibpac Code 25 5
San Diego, CA 93155

Librarian, Naval Intelligence
Support Center
4301 Suitland Road
Washington, D. C. 20390

Commanding Officer
Naval Civil Engineering Laboratory
Port Hueneme, CA 93041

Chief, Wave Dynamics Division
USAE-WES
P. O. Box 631
Vicksburg, MS 39180

Commandant
U. S. Coast Guard
ATTN: GECV/61
Washington, D. C. 20591

Office of Research and Development
%DS/62
U. S. Coast Guard
Washington, D. C. 20591

National Oceanographic Data
Center %D764
Environmental Data Services
NOAA
Washington, D. C. 20235

Prof. Dr. Fuehrboeter
Lehrstuhl F. Hydromechanik U. Kuestenw
Technische Hochschule Braunschweig
Beethovenstrasse 51A
D-3300 Braunschweig, West Germany

Prof. Dr. Walter Hansen
Direktor D. Instituts F. Meereskunde
Universitaet Hamburg
Heimhuderstrasse 71
D-2000 Hamburg 13, West Germany

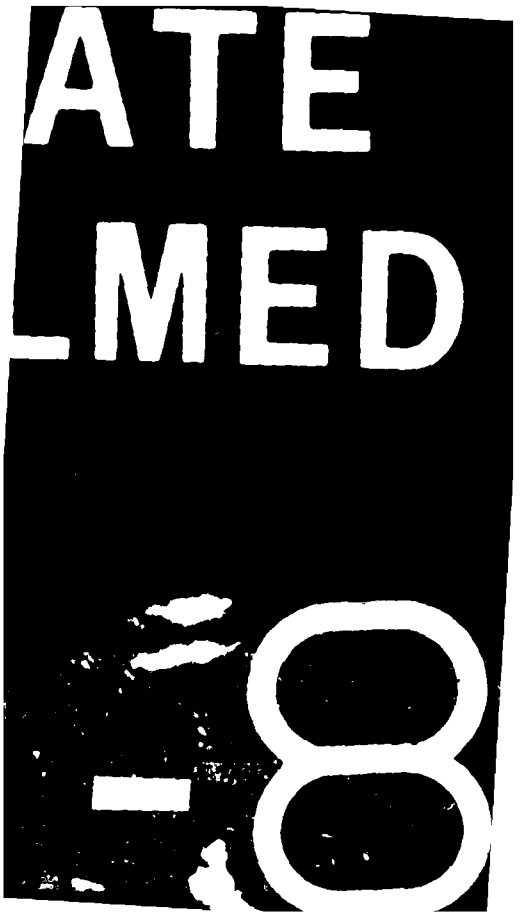
Prof. Dr. Klaus Hasselmann
Institut F. Geophysik
Universitaet Hamburg
Schleuterstrasse 22
D-2000 Hamburg 13, West Germany

Coastal Studies Institute
Louisiana State University
Baton Rouge, LA 70803

Dr. Edward Thornton
Department of Oceanography
Naval Postgraduate School
Monterey, CA 93940

Dr. Douglas I. Inman
University of California A-009
Shore Processes Laboratory
La Jolla, CA 92093

Dr. Bruce Heyden
Dept. of Environmental Sciences
University of Virginia
Charlottesville, VA 22903


ATE
LMED
-8